

Analisi di dati clinici relativi allo screening neonatale esteso e progettazione di metodologie di big data analysis e machine learning per la previsione delle malattie metaboliche nella popolazione pediatrica lombarda

Relatore: Prof. Federico CABITZA

Co-relatore: Dott. Luca MARCONI

Tesi di Laurea Magistrale di:

Andrea Lucini Paioni

Matricola 826578

Anno Accademico 2022/2023

Indice

<i>Indice</i>	3
<i>Abstract</i>	7
<i>1. Il progetto Buzzi</i>	9
1.1.1 Descrizione contesto e screening neonatale esteso	9
1.1.2 Percorso di raccolta dati	11
1.1.3 Descrizione dataset	12
1.1.4 Fasi del progetto	15
<i>2. Pulizia dei dati ed analisi esplorativa</i>	18
 2.1 Metodologia	18
2.1.1 Import e pulizia dei dati	18
2.1.2 Variabili quantitative	18
2.1.3 Correlazioni	19
2.1.4 Variabili qualitative	19
2.1.5 Ulteriori esplorazioni	19
2.1.6 Inferenza	20
 2.2 Risultati e discussione	21
2.2.1 Statistiche descrittive variabili quantitative	21
2.2.2 Grafici variabili quantitative	23
2.2.3 Analisi correlazioni	24
2.2.4 Analisi variabili qualitative	29
2.2.5 Barplot variabili qualitative	35
2.2.6 Esplorazioni ulteriori	38
2.2.7 Inferenza	42
 2.3 Risultati e discussione analisi per dati stratificati	46
2.3.1 Statistiche descrittive variabili quantitative	46
2.3.2 Grafici variabili quantitative	50
2.3.3 Analisi variabili qualitative	50
2.3.4 Barplot variabili qualitative	52
<i>3. La riduzione della dimensionalità</i>	58
3.1.1 Introduzione	58
3.1.2 Cenni storici	58
3.1.3 Obiettivi della riduzione di dimensionalità	58
3.1.4 Parametri chiave nella riduzione di dimensionalità	59
3.1.5 Applicazioni della riduzione di dimensionalità	60
3.1.6 Focus sulle applicazioni mediche e biostatistiche	61
3.1.7 Integrazione con tecniche di cluster analysis	61
 3.2 Principal Component Analysis (PCA)	63
3.2.1 Introduzione	63
3.2.2 Iperparametri e tuning	64
3.2.3 Passaggi algoritmici	64
3.2.4 Vantaggi e svantaggi	64
 3.3 t-distributed Stochastic Neighbor Embedding (t-SNE)	66

3.3.1 Introduzione	66
3.3.2 Iperparametri e tuning	66
3.3.3 Passaggi algoritmici	67
3.3.4 Vantaggi e svantaggi	68
3.4 Uniform Manifold Approximation and Projection (UMAP)	70
3.4.1 Introduzione	70
3.4.2 Iperparametri e tuning	70
3.4.3 Passaggi algoritmici	71
3.4.4 Vantaggi e svantaggi	72
3.5 Altri metodi di riduzione della dimensionalità	73
3.5.1 Kernel Principal Component Analysis (kernel PCA)	73
3.5.2 Independent Component Analysis (ICA)	73
3.5.3 Singular Value Decomposition (SVD)	73
3.5.4 Non-Negative Matrix Factorization (NMF)	74
3.5.5 Criticità dei metodi kernel PCA, ICA, SVD e NMF	74
3.6 Risultati e discussione riduzione dimensionalità	75
3.6.1 UMAP	75
3.6.2 PCA	79
3.6.3 t-SNE	79
4. Cluster Analysis	84
4.1.1 Introduzione	84
4.1.2 Cenni storici	84
4.1.3 Obiettivi della cluster analysis	84
4.1.4 Parametri chiave nella cluster analysis	84
4.1.5 Applicazioni della cluster analysis	85
4.1.6 Focus sulle applicazioni mediche e biostatistiche	86
4.1.7 Integrazione con tecniche di riduzione della dimensionalità	88
4.2 K-means	89
4.2.1 Introduzione	89
4.2.2 Passaggi algoritmici	89
4.2.3 Iperparametri e tuning	90
4.2.4 Il numero ottimale di clusters k: una scelta complessa	91
4.2.5 Vantaggi e svantaggi	91
4.3 Clustering gerarchico	93
4.3.1 Introduzione	93
4.3.2 Dendrogramma	94
4.3.3 Passaggi algoritmici	94
4.3.4 Tipi di distanza	95
4.3.5 Iperparametri e tuning	96
4.3.6 Vantaggi e svantaggi	97
4.4 Density-Based Spatial Clustering of Applications with Noise (DBSCAN)	98
4.4.1 Introduzione	98
4.4.2 Passaggi algoritmici	98
4.4.3 Iperparametri e tuning	99
4.4.4 Vantaggi e svantaggi	99
4.5 BIRCH Clustering	101

4.5.1 Introduzione	101
4.5.2 Passaggi algoritmici	101
4.5.3 Iperparametri e tuning	102
4.5.4 Vantaggi e svantaggi	102
4.6 Spectral Clustering	104
4.6.1 Introduzione	104
4.6.2 Passaggi algoritmo	104
4.6.3 Iperparametri e tuning	105
4.6.4 Vantaggi e svantaggi	105
4.7 Metodi di valutazione delle tecniche di cluster analysis	107
4.8 Risultati e discussione cluster analysis	109
4.8.1 Clustering gerarchico agglomerativo	109
4.8.2 BIRCH	113
4.8.3 DBSCAN	114
4.8.4 K-means	115
4.8.5 Spectral Clustering	116
5. Risultati e discussione cluster analysis dopo riduzione di dimensionalità	118
5.1.1 UMAP	118
5.1.2 PCA	122
5.1.3 t-SNE	125
5.2 Analisi miglior risultato di cluster analysis ottenuto	130
5.2.1 Statistiche descrittive variabili quantitative stratificate per cluster	130
5.2.2 Variabili qualitative	132
6. Conclusioni	135
6.1.1 Sviluppi futuri	136
Appendice	138
Figure analisi esplorative	138
Figure analisi esplorative stratificate per reparto	148
Figure analisi esplorative stratificate per cluster	158
Indice delle figure	165
Indice delle tavole	167
Bibliografia	169
Ringraziamenti	173

Abstract

Il progetto Buzzi, nato all'interno del contesto dell'ospedale Buzzi, e dunque dell'ASST Fatebenefratelli-Sacco, si pone come obiettivo principale di analizzare uno dei più ampi e significativi database in Europa relativi allo screening neonatale esteso (un test, obbligatorio a partire dagli anni 90' su tutto il territorio nazionale italiano, che permette di individuare preventivamente patologie metaboliche e particolari condizioni cliniche nei neonati).

Il lavoro punta ad automatizzare la strutturazione ed elaborazione del dataset relativo a dati clinici e demografici di tutti i neonati, sani o con delle variazioni nelle analisi biochimiche considerate, nati in Regione Lombardia a partire da Giugno 2012 fino ad arrivare ad Aprile 2022, all'interno di un contesto fortemente human-centric e ad alta eterogeneità informativa.

L'obiettivo è di identificare, progettare, realizzare e valutare un set esteso di metodologie sinergiche o complementari di big data analysis e machine learning per l'analisi dei dati, col fine della modellazione predittiva in ambito clinico. L'analisi dei dati raccolti attraverso questo screening può essere utilizzata per individuare eventuali correlazioni, patterns o signatures all'interno dei dati raccolti, e prevedere la comparsa di malattie metaboliche nella popolazione pediatrica lombarda.

Lo sviluppo di questo progetto, a partire dalle basi teoriche, passando per lo sviluppo di algoritmi fino ad arrivare alla fase di ricerca nella letteratura scientifica di progetti corrispondenti, è stato svolto all'interno di un percorso di stage curricolare presso il MUDI Lab dell'Università degli studi di Milano – Bicocca: in questo contesto ho avuto modo di acquisire esperienza nell'analisi di big data relativi all'ambito medico e clinico, grazie alla guida dei miei tutor, il Prof. Federico Cabitza e il Dott. Luca Marconi, e al supporto degli altri membri del MUDI Lab. Nei mesi di attività e sviluppo del lavoro sono state approfondite (dal punto di vista sia teorico che pratico) tecniche di analisi dei dati e algoritmi di machine Learning, col fine di indagare eventuali patterns presenti all'interno dei dati relativi alla popolazione pediatrica lombarda.

Nel concreto, tramite codice Python, è stato possibile procedere con l'importazione dei dati, una fase di pulizia degli stessi, una fase di data exploration (con l'utilizzo di numerosi grafici, statistiche descrittive, metodi di inferenza, analisi di correlazione e analisi dei pazienti stratificati per ottenere insights su variabili di particolare interesse all'interno del dataset), una fase di riduzione di dimensionalità (dove sono state applicate diverse metodologie, tra cui UMAP, PCA e t-SNE), una fase di cluster analysis (su dati con o senza applicazione di metodi di riduzione di dimensionalità, e tramite l'utilizzo di tecniche quali il clustering gerarchico agglomerativo, il BIRCH, il DBSCAN e il K-means), ed un'analisi approfondita dei risultati ottenuti con le tecniche citate.

1. Il progetto Buzzi

1.1.1 Descrizione contesto e screening neonatale esteso

L’Ospedale dei Bambini Vittore Buzzi fa parte dell’ASST (Azienda Socio Sanitaria Territoriale) Fatebenefratelli-Sacco, nata il 1° gennaio 2016 con l’unione di quattro presidi Ospedalieri (Ospedale dei Bambini Vittore Buzzi, Ospedale Fatebenefratelli e Oftalmico, Ospedale Luigi Sacco - Polo Universitario, Ospedale Macedonio Melloni) e di 27 sedi territoriali dei Municipi 1, 2, 3, 4 e 8 della città di Milano.

In particolare, l’Ospedale dei Bambini Vittore Buzzi, è un ospedale pediatrico di Milano fondato nel 1906 a partire da un piccolo ospedale da 12/14 posti letto per bambini tra i 2 e gli 8 anni; si tratta di un progetto avviato dal medico Raimondo Guaita verso la fine del XIX secolo, assieme ad un gruppo di promotori che lo aiutarono per la fase di raccolta fondi. L’attività inizia a partire dal 1895 all’interno dello stabile delle Suore di Maria Bambina in via San Vincenzo 25, per poi passare, nel dicembre 1906, col completamento dell’edificio, nella sede attuale situata in via Castelvetro n.32. Il nome dell’ospedale deriva dall’imprenditore, filantropo e studioso di toponomastica italiano Vittore Buzzi, che con una ricca donazione permise nel 1967 la costruzione del terzo blocco dell’ospedale.

Il progetto Buzzi è un progetto nato proprio all’interno del contesto dell’ospedale Buzzi, con l’obiettivo di analizzare uno dei più ampi e significativi database in Europa relativi allo screening neonatale esteso. Si tratta di un dataset relativo a dati clinici e demografici di tutti i neonati, sani o con delle variazioni nelle analisi biochimiche considerate, nati in Regione Lombardia a partire da Giugno 2012 fino ad arrivare ad Aprile 2022.

Il lavoro punta ad automatizzare la strutturazione ed elaborazione del patrimonio informativo clinico lombardo, all’interno di un contesto fortemente human-centric e ad alta eterogeneità informativa.

L’obiettivo è dunque di identificare, progettare, realizzare e valutare un set esteso di metodologie sinergiche o complementari di big data analysis e machine learning per l’analisi dei dati, col fine della modellazione predittiva in ambito clinico. L’analisi dei dati raccolti attraverso questo screening può essere utilizzata per individuare eventuali correlazioni, patterns o signatures all’interno dei dati raccolti, e prevedere la comparsa di malattie metaboliche nella popolazione pediatrica lombarda.

Al momento, non è presente nel Database l’informazione relativa a quali di questi neonati siano risultati poi effettivamente affetti da patologia metabolica. L’obiettivo attuale, dunque, riguarda fasi preliminari all’effettiva valutazione dell’insorgenza o meno di queste patologie.

Lo screening neonatale esteso è un test, nato all’interno di un programma di prevenzione obbligatorio a partire dal 1992 su tutto il territorio nazionale italiano, che permette di individuare patologie e condizioni nel neonato e di conseguenza di mettere in atto terapie in maniera tempestiva. Lo screening neonatale nasce con l’obiettivo di individuare preventivamente una serie di malattie metaboliche, tra cui:

- **Ipotiroidismo congenito (IC):** si tratta di una delle più frequenti endocrinopatie dell’età evolutiva, con un’incidenza che varia tra 1:3000 e 1:1200 nati vivi; può essere causato da difetti funzionali della tiroide (formata normalmente e con corretta collocazione nella sua sede naturale alla base del collo) o più frequentemente da alterazioni nella embriogenesi della ghiandola tiroidea (con assenza della ghiandola stessa, con ipoplasia o con la presenza di un abbozzo tiroideo in sede ectopica), generalmente insufficiente ad assicurare un normale apporto di ormoni tiroidei.

Si tratta di una condizione permanente o transitoria (spesso causata da fattori materni o neonatali), con effetti particolarmente gravi sul sistema nervoso centrale, e che quindi determina spesso forme di ritardo neurocognitivo.

Un'adeguata terapia ormonale sostitutiva (L-tiroxina), poco costosa e di semplice somministrazione, consente di prevenire tali danni purché sia attuata precocemente.

- **Fibrosi cistica (o mucoviscidosi):** malattia presente dalla nascita dovuta ad una mutazione del gene CFTR, ereditata da padri e madri (portatori sani; in Italia viene stimato un portatore sano ogni 30 persone circa) entrambi con copie di gene mutato; ad ogni gravidanza, la coppia di portatori sani ha una probabilità su quattro di avere un figlio malato.

La fibrosi cistica altera le secrezioni di molti organi che, risultando più dense, disidratate e poco fluide, contribuiscono al loro danneggiamento: a subire il maggior danno sono solitamente bronchi e polmoni (al loro interno il muco tende a ristagnare, generando infezioni ed infiammazioni) oltre al pancreas; queste infiammazioni, nel tempo, tendono a portare ad insufficienze respiratorie, mentre il progredire del danno pancreatico porta spesso a forme di diabete.

Oggi ci sono più adulti che bambini con fibrosi cistica, e questi conducono una vita normale; tuttavia le statistiche suggeriscono un'aspettativa mediaiana intorno ai 40 anni di vita (previsioni comunque in continuo miglioramento grazie ai progressi della ricerca).

Di recente sono stati scoperti farmaci che intervengono efficacemente su alcuni tipi di mutazione del gene CFTR, in modo da bloccare sul nascere la malattia e rendere più efficaci le cure di cui già disponiamo.

- **Fenilchetonuria (PKU):** raro difetto metabolico ereditario (interessa circa 50.000 persone in tutto il mondo) a trasmissione autosomica recessiva; questa patologia porta a disabilità intellettuale, microcefalia, deficit motori, disturbi dello spettro autistico, epilessia, ritardo dello sviluppo, deficit di accrescimento e sintomi psichiatrici. Se non adeguatamente trattata, la patologia comporta un grave e irreversibile ritardo mentale, oltre a importanti disabilità cognitive.

Il trattamento si basa prevalentemente su un rigoroso regime alimentare a basso contenuto di proteine, che i pazienti devono seguire per tutta la vita.

- **Iperplasia surrenale congenita:** gruppo di rare malattie genetiche (nelle forme più frequenti, un caso ogni 16000 nati), ognuna caratterizzata da un'insufficiente produzione di cortisolo, aldosterone o entrambi; sospettata nei neonati con grave deficit dell'accrescimento ponderale, e in caso di ipertrofia del clitoride.

Il trattamento farmacologico consiste nel correggere il deficit di glucorticoidi e mineralcorticoidi; nell'adulto il trattamento deve continuare per evitare l'iperandrogenismo, per mantenere il normale ciclo mestruale e la fertilità nelle donne. Il trattamento chirurgico, invece, consiste nella genito plastica femminilizzante, per ricreare l'anatomia dei genitali esterni.

- **Atrofia Muscolare Spinale (SMA):** malattia neuromuscolare rara (un'incidenza di circa 1 paziente su 10mila nati vivi) caratterizzata dalla perdita dei motoneuroni (i neuroni che trasportano i segnali dal sistema nervoso centrale ai muscoli, controllandone il movimento); provoca debolezza e atrofia muscolare progressiva che interessa gli arti inferiori e i muscoli respiratori.

Nel 95% dei casi, la patologia è causata da specifiche mutazioni nel gene SMN1. Sono state approvate tre terapie specifiche per questa malattia: *l'oligonucleotide antisenso nusinersen*, il farmaco orale *risdiplam* e la terapia genica *onasemnogene abeparvovec*. Alcuni dati preliminari indicano che i bambini trattati prima della comparsa dei sintomi presentano uno sviluppo motorio equiparabile a quello dei bambini non affetti.

Lo screening avviene attraverso il prelievo di poche gocce di sangue dal tallone del neonato, effettuato tra le 48 e le 72 ore di vita; i campioni sono poi analizzati nel Laboratorio specializzato per lo Screening e i risultati delle analisi vengono poi inviati all'Ospedale di nascita. In caso di risultato positivo, i genitori vengono contattati dal Personale sanitario dell'Ospedale di nascita, che provvede a fornire alla famiglia ogni informazione necessaria. Un risultato positivo di un test di screening non significa necessariamente che il bambino sia ammalato, ma rende necessaria l'esecuzione di esami d'approfondimento diagnostico.

Ci sono stati nel corso del tempo dei cambiamenti di tecnologia per i macchinari utilizzati nello screening neonatale, in particolare la sostituzione della precedente strumentazione AutoDELFIA® immunoassay system for prenatal screening (PerkinElmer) con le nuove GSP® Instrument (PerkinElmer) ed il cambiamento degli spettrometri di massa. Nello studio non sono state rilevate differenze significative col cambio di macchinari, ma la verifica dell'impatto di questi cambiamenti sull'attività analitica è una possibile domanda di ricerca futura.

1.1.2 Percorso di raccolta dati

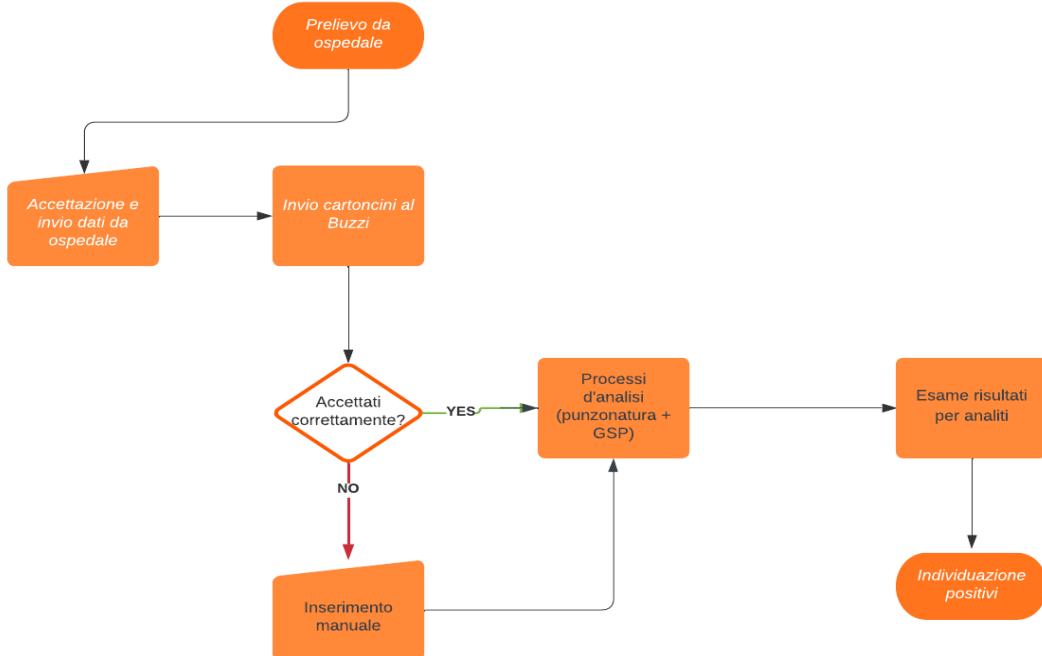


Figure 1.1: flowchart procedimento di raccolta dati

Proviamo ora a fare chiarezza sulle fasi del processo di acquisizione dei dati, come descritto brevemente dallo schema in alto nella Figura [1.1]:

- Raccolta ed “accettazione” (inserimento) dei dati nel sistema:** per ogni paziente (dunque per ogni neonato testato) e in ogni singolo ospedale, vengono raccolte le necessarie gocce di sangue e viene compilato il relativo cartoncino, detto “cartoncino di Guthrie”. Sul cartoncino vengono riportati i dati demografico-anamnestici del paziente e vengono depositate le gocce di sangue. Successivamente, viene attuata l’accettazione, ossia l’inserimento dei dati nel sistema, dal singolo ospedale di provenienza del campione.
Quest’operazione può presentare una serie di problemi, legati principalmente alla compilazione dei cartoncini (errori relativi a dati demografico-anamnestici non corretti e/o non completi, oppure errori relativi alla tipologia di prelievo effettuata; questi errori sono corretti dall’operatore al Buzzi, una volta che i cartoncini sono stati ricevuti), alla raccolta e deposito delle gocce di sangue (possono dare origine a misurazioni non corrette o non coerenti, anche in questo caso verificate dall’operatore al Buzzi che, se necessario, fa dei retest o richiede un nuovo prelievo).
- Invio dei cartoncini al Buzzi:** i cartoncini vengono inviati e divisi in “pacchi” numerati dai singoli ospedali periodicamente. Gli operatori del Buzzi hanno definito una numerazione di comodo ed i cartoncini vengono identificati per numero del pacco, data e numero di cartoncino.
Anche in questa fase possono esserci alcuni potenziali errori legati al procedimento, ovvero relativi allo smarrimento dei cartoncini da parte dei commessi (sarebbe ovviamente ottimale definire una modalità di trasmissione alternativa dei campioni, per evitare o ridurre significativamente i casi di smarrimento di dati).

3. Analisi dei cartoncini al Buzzi: gli operatori inseriscono i cartoncini nelle macchine puncher (da “punzonatura”), che lavorano il cartone dove sono depositate le gocce di sangue. Successivamente, i cartoncini vengono inseriti nelle macchine GSP di PerkinElmer, caricate con i necessari reagenti, che misurano i diversi analiti.

Nelle piastre di analisi vengono anche caricate le necessarie calibrazioni sia per gli analiti misurati in immunometria di massa (per cui sono già presenti le rette di calibrazione) sia per gli analiti misurati in spettrometria di massa (in questo caso non ci sono delle rette di calibrazione, ma delle sostanze “simili” a quelle da misurare, ad esempio con un loro isotopo al posto dell’idrogeno, in modo da essere usate come confronti per la calibrazione).

Anche in questa fase ci possono essere alcuni problemi legati alla gestione della periodicità del cambio dei reagenti, che generalmente vengono mantenuti per qualche giorno (non è infatti chiarissimo quale sia la periodicità di cambio ottimale); possibili errori di misurazione da parte dei macchinari invece non sono stati segnalati dagli operatori.

Gli intervalli di riferimento (IR) usati dal laboratorio sono calcolati utilizzando un software Cat-off Analyzer sui dati prodotti dal laboratorio stesso usando i percentili. Il software considera un parametro alla volta, rendendo molto macchinoso il calcolo e l’aggiornamento frequente degli IR. Infatti questi devono essere differenziati per alcuni analiti per quattro diverse classi di peso ($weight < 1500$ gr; $1500\text{gr} < weight < 2000$ gr; 2000 gr $< weight < 2500$ gr; $weight > 2500$ gr) e per i tre tempi di prelievo elencati sotto (prelievo basale, prelievo bis o prelievo tris).

4. Analisi dei risultati e individuazione dei positivi: i risultati degli analiti da analizzare sono divisi tra i vari operatori del Buzzi, che verificano se i valori rientrano o meno nei corretti intervalli di riferimento, per accertare o meno la positività di un paziente rispetto alle diverse patologie. Laddove sia necessario un retest, il sistema informativo lo segnala per l’analita corrispondente.

Gli operatori possono comunque richiedere un retest di tutti gli analiti, se i valori non convincono. Il retest viene fatto in duplicato (“in doppio”). Verosimilmente il sistema ha già dato le media dei 2 valori di retest e quello che ci si ritrova è il valore sulla prima macchia e la media degli altri 2.

Per ogni neonato è previsto un prelievo basale tra le 48-72 ore dalla nascita mentre, al verificarsi di determinate condizioni, si possono effettuare altri due possibili prelievi:

- **Prelievo basale** 48-72 ore dalla nascita, viene eseguito su tutti i neonati.
- **Prelievo bis** a 15 gg dalla nascita, viene effettuato solo per alcune condizioni materne o neonatali (ad esempio, mamma con patologia tiroidea pregressa, trattamento cortisonico neonatale o materno, neonato nato sotto le 37 settimane di età gestazionale, neonato nato sotto i 2000 gr di peso, gemellarità, ricovero in terapia intensiva...).
- **Prelievo tris** a 28 gg dalla nascita, viene effettuato solo per neonati nati sotto i 2000 gr di peso o sotto le 34 settimane di età gestazionale.

1.1.3 Descrizione dataset

Il dataset ottenuto a partire dai passaggi descritti nel paragrafo precedente [1.1.2] comprende dunque tutti i dati di bambini nati, sani o con variazioni nelle analisi biochimiche considerate, nella Regione Lombardia da circa metà 2012 fino all’Aprile 2022, per un totale di 985792 records.

Le features riportate inizialmente nel dataset sono in totale 266, di cui 235 riferite ad analiti vari e 31 riferite a caratteristiche del paziente, ossia variabili di anagrafica e di dati clinici (peso, età gestazionale, etnia, sesso, dieta materna e neonatale ecc.). Tuttavia, alcuni analiti sono stati esclusi dalla fase di analisi per una serie di considerazioni: non solo alcuni sono stati considerati non rilevanti dai clinici esperti di dominio, ma alcuni sono stati rilevati solo per una piccola percentuale di pazienti; inoltre è stato deciso dai clinici di

escluderne alcuni considerati superflui, o comunque poco significativi per lo studio; infine una serie di features sono state eliminate dal dataset grazie ad analisi preliminari di collinearità effettuate sui dati.

Dunque, il dataset finale considerato è formato da 985792 osservazioni (in particolare, per i fini di questo lavoro, è stato considerato un campione di circa il 30% delle osservazioni, per un totale di 295738 occorrenze) con un totale di 107 features considerate. Di queste:

- 74 variabili quantitative sono di tipo decimale, e si tratta degli analiti:
 - *ASATotal* (acido argininosuccinico, unità di misura $\mu\text{mol/L}$)
 - *Ala* (alanina, unità di misura $\mu\text{mol/L}$)
 - *Allele 1* (risultato di un test genetico che rileva la presenza o l'assenza di una determinata variante genetica, allele, associata a una malattia ereditaria, unità di misura differente in base al risultato)
 - *Arg* (arginina, unità di misura $\mu\text{mol/L}$)
 - *Cit* (citrullina, unità di misura $\mu\text{mol/L}$)
 - *Glu* (acido glutammico, unità di misura $\mu\text{mol/L}$)
 - *Gly* (glicina, unità di misura $\mu\text{mol/L}$)
 - *Leu\Ile\Pro-OH* (leucina\isoleucina\idrossiprolina, unità di misura $\mu\text{mol/L}$)
 - *MET* (metionina, unità di misura $\mu\text{mol/L}$)
 - *Orn* (ornitina, unità di misura $\mu\text{mol/L}$)
 - *PHE* (fenilalanina, unità di misura $\mu\text{mol/L}$)
 - *Pro* (prolina, unità di misura $\mu\text{mol/L}$)
 - *TYR* (tirosina, unità di misura $\mu\text{mol/L}$)
 - *Val* (valina, unità di misura $\mu\text{mol/L}$)
 - *BTD* (human biotinidase activity, unità di misura U/dl)
 - *C0* (carnitina libera, unità di misura $\mu\text{mol/L}$)
 - *C3* (propionilcarnitina, unità di misura $\mu\text{mol/L}$)
 - *C3DC\C4OH* (malonilcarnitina\3-idrossi-butirilcarnitina, unità di misura $\mu\text{mol/L}$)
 - *C4* (butirilcarnitina, unità di misura $\mu\text{mol/L}$)
 - *C4DC\C5OH* (metilmalonile\3-idrossi-isovalerilcarnitina, unità di misura $\mu\text{mol/L}$)
 - *C5* (isovalerilcarnitina, unità di misura $\mu\text{mol/L}$)
 - *C5:1* (tiglilcarnitina, unità di misura $\mu\text{mol/L}$)
 - *C5DC\C6OH* (glutarilcarnitina\3-idrossi-esanoilcarnitina, unità di misura $\mu\text{mol/L}$)
 - *C6* (esanoilcarnitina, unità di misura $\mu\text{mol/L}$)
 - *C6DC* (adipilcarnitina, unità di misura $\mu\text{mol/L}$)
 - *C8* (ottanoilcarnitina, unità di misura $\mu\text{mol/L}$)
 - *C8:1* (octenoilcarnitina, unità di misura $\mu\text{mol/L}$)
 - *C10* (decanoilcarnitina, unità di misura $\mu\text{mol/L}$)
 - *C10:1* (decenoilcarnitina, unità di misura $\mu\text{mol/L}$)
 - *C10:2* (decadienoilcarnitina, unità di misura $\mu\text{mol/L}$)
 - *C12* (dodecanoilcarnitina, unità di misura $\mu\text{mol/L}$)
 - *C12:1* (dodecenoilcarnitina, unità di misura $\mu\text{mol/L}$)
 - *C14* (tetradecanoilcarnitina, o miristoilcarnitina, unità di misura $\mu\text{mol/L}$)
 - *C14:1* (tetradecenoilcarnitina, unità di misura $\mu\text{mol/L}$)
 - *C14:2* (tetradecadienoilcarnitina, unità di misura $\mu\text{mol/L}$)
 - *C14-OH* (3-idrossi-tetradecanoilcarnitina, unità di misura $\mu\text{mol/L}$)
 - *C16* (esadecanoilcarnitina, unità di misura $\mu\text{mol/L}$)
 - *C16:1* (esadecenoilcarnitina, unità di misura $\mu\text{mol/L}$)
 - *C16-OH* (3-idrossi-esadecanoilcarnitina, unità di misura $\mu\text{mol/L}$)
 - *C16:1-OH* (3-idrossi-esadecenoilcarnitina, unità di misura $\mu\text{mol/L}$)

-
- *C18* (ottadecanoilcarnitina, o stearoilcarnitina, unità di misura $\mu\text{mol/L}$)
 - *C18:1* (ottadecenoilcarnitina, o oleoilcarnitina, unità di misura $\mu\text{mol/L}$)
 - *C18:2* (ottadecadienoilcarnitina, o linoleoilcarnitina, unità di misura $\mu\text{mol/L}$)
 - *C18-OH* (3-idrossi-ottadecanoilcarnitina, unità di misura $\mu\text{mol/L}$)
 - *C18:1-OH* (3-idrossi-ottadecenoilcarnitina, unità di misura $\mu\text{mol/L}$)
 - *C18:2OH* (3-idrossi-ottadecadienoilcarnitina, unità di misura $\mu\text{mol/L}$)
 - *C20* (eicosanoilcarnitina, o arachidoilcarnitina, unità di misura $\mu\text{mol/L}$)
 - *C22* (docosanoilcarnitina, o beenonilcarnitina, unità di misura $\mu\text{mol/L}$)
 - *C24* (tetracosanoilcarnitina, o lignoceroilcarnitina, unità di misura $\mu\text{mol/L}$)
 - *C26* (esacosanoilcarnitinam, o cerotolcarnitina, unità di misura $\mu\text{mol/L}$)
 - *SA* (succinilacetone, unità di misura $\mu\text{mol/L}$)
 - *ADO* (adenosina, unità di misura $\mu\text{mol/L}$)
 - *D-ADO* (2'-deossiadenosina, unità di misura $\mu\text{mol/L}$)
 - *C20:0-LPC* (*C20:0* lisofosfatidilcolina, unità di misura $\mu\text{mol/L}$)
 - *C22:0-LPC* (*C22:0* lisofosfatidilcolina, unità di misura $\mu\text{mol/L}$)
 - *C24:0-LPC* (*C24:0* lisofosfatidilcolina, unità di misura $\mu\text{mol/L}$)
 - *C26:0-LPC* (*C26:0* lisofosfatidilcolina, unità di misura $\mu\text{mol/L}$)
 - *s-TSH* (human thyroid stimulating hormone, unità di misura $\mu\text{U/mL}$)
 - *IRT-GSP* (human immunoreactive trypsin, unità di misura ng/mL)
 - *TGAL* (total galactose, galactose and galactose 1-phosphate, unità di misura mg/dL)
 - *MMA* (acido metilmalicino, unità di misura μM)
 - *EMA* (acido etilmalicino, unità di misura μM)
 - *GA* (acido glutarico, unità di misura μM)
 - *2OH GA* (acido 2-OH glutarico, unità di misura μM)
 - *3OH GA* (acido 3-OH glutarico, unità di misura μM)
 - *HCYs* (omocisteina, unità di misura μM)
 - *3OH PA* (acido 3-OH propionico, unità di misura μM)
 - *MCA* (acido metilcitrico, unità di misura μM)
 - *OROTICO* (acido orotico, unità di misura μM)
 - *PIVA* (pivaloilcarnitina, unità di misura μM)
 - *2MBC* (2-metilbutirrilcarnitina, unità di misura μM)
 - *c4-b* (butirrilcarnitina, unità di misura μM)
 - *c4-i* (isobutirrilcarnitina, unità di misura μM)
- 2 sono variabili quantitative di tipo intero, ovvero la variabile *GestationalAge* (che indica il numero di settimane intercorse tra l'ultimo ciclo mestruale prima del concepimento e il giorno del parto), e la variabile *Weight* (ovvero il peso del neonato al momento della nascita)
 - 26 variabili sono di tipo qualitativo, ed indicano una serie di informazioni relative al parto, ai neonati, alle madri e alla raccolta dei campioni:
 - *Sex* (Sesso, categoria, come "M" o "F")
 - *City* (città di residenza, in stringa)
 - *Sampling* (tipo di prelievo, categoria, come "Iniziale", "BIS" o "Controllo")
 - *Etnicity* (origine geografica, in stringa, con 149 differenti nazionalità o etnie)
 - *SampleQuality* (qualità del campione, categoria, come "inadeguata", "insufficiente" o "ok")
 - *AntibioticsBaby* (assunzione di antibiotici per il bambino, categoria, "0" per "no" o "1" per "sì")
 - *AntibioticsMother* (assunzione di antibiotici per la madre, categoria, "0" per "no" o "1" per "sì")
 - *Meconium* (ileo meconio, categoria, , "0" per "no" o "1" per "sì")
 - *CortisoneBaby* (assunzione di cortisone per il bambino, categoria, "0" per "no" o "1" per "sì")
 - *CortisoneMother* (assunzione di cortisone per la madre, categoria, "0" per "no" o "1" per "sì")

- *TyroidMother* (patologia tiroidea della madre, categoria, “0” per “no” o “1” per “sì”)
- *Premature* (bambino prematuro, categoria, “0” per “no” o “1” per “sì”)
- *TooYoung* (prelievo prima delle 48 ore, categoria, “0” per “no” o “1” per “sì”)
- *BabyFed* (bambino alimentato naturalmente, categoria, “0” per “no” o “1” per “sì”)
- *HUFeed* (alimentazione materna, categoria, “0” per “no” o “1” per “sì”)
- *MIXFeed* (alimentazione materna o altro, categoria, “0” per “no” o “1” per “sì”)
- *ARTFeed* (alimentazione artificiale, categoria, “0” per “no” o “1” per “sì”)
- *TPNFeed* (alimentazione parenterale, categoria, “0” per “no” o “1” per “sì”)
- *ENFeed* (alimentazione enterale, categoria, “0” per “no” o “1” per “sì”)
- *TPNCARNFeed* (alimentazione parenterale con carnitina, categoria, “0” per “no” o “1” per “sì”)
- *TPNMCTFeed* (alimentazione parenterale con MCT, categoria, “0” per “no” o “1” per “sì”)
- *Hospital* (nome ospedale richiedente screening, stringa)
- *AnswerIX* (fase dell’analisi, stringa, legata alla variabile Sampling)
- *BirthMethod* (metodo di nascita, categoria, “naturale” o “cesareo”)
- *BIS* (campione BIS o no, categoria, “0” per “no” o “1” per “sì”)
- *Twins* (parto gemellare, categoria, “0” per “no” o “1” per “sì”)
- 3 variabili in formato data: le variabili *SamTimeCollected* e *SamTimeReceived*, che indicano rispettivamente data e ora del campione raccolto per ogni neonato e data ed ora dell’invio dei dati da parte dell’ospedale al Buzzi, oltre a *DateOfBirth* (ovvero la data di nascita del bambino).
- 2 variabili di riconoscimento del paziente e del campione, come stringa, ovvero *SampleBarcode* (identifica univocamente il campione) e *ID* (identifica univocamente il neonato).

Infine sono state create due variabili qualitative (che portano il totale di features a 109, di cui 28 qualitative) a partire da variabili già presenti all’interno del dataset:

- La variabile *Reparto*, a partire dalla feature *Hospital* ricodificata, che definisce la provenienza del campione (se da un reparto *Nido*, un reparto *Neo-Patologico* o un reparto *Generico*); si tratta di un aspetto molto importante, perché il reparto del neonato in molti casi può essere indicativo di problematiche pre o post parto, e quindi può essere legato a malattie metaboliche o a sintomi delle stesse.
- La variabile *Etnia*, a partire dalla feature *Etnicity* (che è composta da 149 differenti etnie o nazionalità, non sempre definite in modo univoco e chiaro per la differenza di terminologia utilizzata in ogni singolo ospedale), che permette di ridefinire l’etnia delle madri dei neonati in 7 differenti macro-categorie: *Caucasian, Arab, Asian, Native Hawaiian or Other Pacific Islander, Hispanic/Latino, Black or African American e Other*.

1.1.4 Fasi del progetto

Trattandosi di un progetto ancora in fase di sviluppo, i differenti steps del progetto (tutti effettuati utilizzando il linguaggio di programmazione *Python*) sono stati concordati all’interno del gruppo di lavoro, con il supporto dei clinici dell’Ospedale Buzzi e i membri del gruppo di lavoro del MUDI (Model Uncertainty, Decisions and Interactions) Lab all’interno del DISCO - Dipartimento di Informatica, Sistemistica e Comunicazione - dell’Università degli Studi di Milano - Bicocca:

1. **Fase di import e pulizia dei dati:** si tratta di una fase fondamentale in ogni progetto di ricerca e di data science; avendo un dataset di dimensioni importanti, con determinati requisiti di privacy, data la delicatezza del contesto e dei dati forniti, e considerata la natura e gli obiettivi dello studio, è fondamentale poter lavorare con dati puliti e validi; inoltre il codice dev’essere riproducibile anche nel caso in cui i dati disponibili fossero integrati sia da features aggiuntive (come informazioni

sull’insorgenza di malattie metaboliche) sia da nuovi dati (come l’aggiunta di dati successivi al mese di Aprile 2022).

2. **Fase di analisi esplorativa:** alla prima fase di import e pulizia dei dati segue la fase di analisi esplorativa, dove l’obiettivo è stato acquisire dimestichezza con i dati, ottenendo informazioni sulla struttura del dataset, sulle caratteristiche e le distribuzioni delle variabili, analizzando la presenza o assenza di collinearità tra le diverse features del dataset, ed ottenendo informazioni aggiuntive su come sono distribuite congiuntamente una serie di variabili di particolare interesse (il tutto sia tramite calcolo di indicatori statistici, sia utilizzando tecniche di visualizzazione grafica delle distribuzioni delle variabili).
3. **Fase di analisi esplorativa stratificata per la variabile “Reparto”:** fase simile alla fase n°2, ma con un focus particolare sulle differenze ottenute rispetto alla precedente stratificando per una variabile considerata di particolare importanza, ovvero la variabile “Reparto”.
4. **Fase di applicazione delle tecniche di riduzione della dimensionalità:** la fase di riduzione della dimensionalità può essere considerata una fase preliminare a quella della cluster analysis applicata a dati ridotti, poiché permette non solo di ottenere risultati validi e significativi riducendo i tempi di esecuzione degli algoritmi, ma anche di ottenere risultati grafici a partire dalle tecniche utilizzate.
5. **Fase di applicazione delle tecniche di cluster analysis:** si tratta di una fase fondamentale per verificare la presenza di gruppi di neonati con caratteristiche simili, ma al contempo si tratta comunque di una fase “esplorativa” per la cluster analysis perché permette di individuare i metodi che restituiscono i risultati più significativi, con tempi di esecuzione e requisiti computazionali minori.
6. **Fase di applicazione delle tecniche di cluster analysis ai dati ridotti dopo l’applicazione di tecniche di riduzione della dimensionalità:** si tratta della fase più importante, perché unendo le informazioni ottenute con gli step precedenti ci si può focalizzare ora su tecniche di cluster analysis efficaci (trovate nella fase 5) applicate a metodi di riduzione della dimensionalità utili e rilevanti (ottenuti con la fase 4), in modo da ottenere risultati significativi.
7. **Valutazione risultati ottenuti:** è la fase in cui vengono valutati in termini di metriche, di chiarezza ed interpretazione i risultati ottenuti; è una fase fondamentale perché da questa nascono poi molti degli spunti di ricerca futuri che potranno essere analizzati.

2. Pulizia dei dati ed analisi esplorativa

Nella prima fase di pulizia del dataset ed analisi esplorativa, il focus principale è stato rendere il dataset pulito, leggibile e pronto ad essere analizzato ed esplorato con metodi ed algoritmi del linguaggio di programmazione *Python*.

2.1 Metodologia

Si possono distinguere diversi passaggi nella fase di esplorazione del dataset. Tutti i seguenti steps sono stati eseguiti sul campione del dataset originale fornito dall’Ospedale Buzzi al laboratorio di ricerca, formato da circa il 30% delle osservazioni iniziali (295738 righe).

Tutti i seguenti passaggi sono stati eseguiti con linguaggio di programmazione *Python*, sull’editor di codice *Visual Studio Code*, col fine di lavorare su ambienti di lavoro in locale - e non in cloud - per avere sempre particolare attenzione relativamente alla sicurezza e privacy dei dati molto sensibili su cui viene svolto l’intero progetto.

2.1.1 Import e pulizia dei dati

Le fasi iniziali per importare e pulire il dataset sono state svolte tramite alcune librerie fondamentali contenute all’interno dell’ambiente di lavoro *Python*: si tratta di packages quali *pandas*, *os*, *scipy* e *numpy* [pandas.pydata.org].

In questa fase iniziale è stato possibile esplorare velocemente la struttura del dataset, osservando il numero e la tipologia di variabili ma anche la dimensione del campione a disposizione; inoltre, sono state create le due variabili *Reparto* ed *Etnia*, fondamentali nelle fasi successive del progetto, come già descritto nel capitolo [1.1.3].

Alla fine di questi passaggi iniziali, il dataset era formato da 295737 osservazioni, con 109 features.

2.1.2 Variabili quantitative

Tramite l’utilizzo del metodo *.describe()* sono state calcolate le statistiche descrittive [[Cicchitelli 17](#)] relative alle 76 variabili quantitative (sono state considerate sia le variabili intere che quelle in formato decimale) presenti all’interno del dataset.

In particolare, vengono calcolati tramite il metodo *.describe()* media, deviazione standard, valori minimo e massimo, valori dei quartili (il secondo quartile è la mediana), e percentuale di dati mancanti, oltre ad una serie di statistiche non presenti nel metodo: indice di curtosi e skewness, la percentuale di outliers (considerati come i valori inferiori al 2.5% quantile e superiori al 97.5% quantile), il range interquartile (chiamato iqr nelle tabelle risultanti, ovvero il range che intercorre tra il valore del terzo e del primo quartile), i due quantili 2.5% e 97.5%, e l’unità di misura delle variabili.

All’interno dell’analisi delle statistiche descrittive, inoltre, sono stati inseriti i grafici [[python-graph-gallery.com](#)] relativi alle distribuzioni delle singole variabili, con un istogramma per ogni variabile (in ascissa il valore assunto dalla variabile, e in ordinata la frequenza per ogni classe).

2.1.3 Correlazioni

Sempre relativamente alle variabili quantitative, tramite il metodo `.corr()` è stato possibile trovare la matrice di correlazione [Cicchitelli 17], che permette di individuare l'eventuale presenza/assenza di correlazione lineare. Allo stesso modo, dopo aver individuato la matrice di correlazione, è stata creata la relativa heatmap che mostra i valori di correlazione tra le variabili quantitative presenti nel dataset.

Un ulteriore passaggio effettuato riguarda il fatto di considerare solamente le variabili quantitative con un numero esiguo di valori mancanti: sono state considerate solamente le variabili con almeno il 90% dei dati non mancanti, ovvero 47 delle 76 variabili quantitative originali; di queste, è stata individuata la matrice di correlazione, e l'heatmap corrispettiva.

Infine, è stata individuata anche una tabella che riportava solamente valori della correlazione elevati (>0.5): in questo modo è possibile individuare eventuali variabili potenzialmente collineari, e in generale variabili che hanno forte correlazione positiva o negativa.

La condizione di collinearità è stata successivamente testata col calcolo del VIF (Variance Inflation Factor, ovvero il Fattore di Inflazione della Varianza), un indice statistico che indica quanto aumenta maggiormente la varianza di una variabile rispetto al fatto se fosse incorrelata con le altre variabili; la formula dell'indice è la seguente:

$$VIF = \frac{1}{(1 - R^2)}$$

con R^2 pari all'R-quadro ottenuto dalla regressione in cui viene considerata come variabile dipendente ogni variabile per cui viene calcolato l'indice.

A valori del VIF superiori a 10 corrispondono variabili che andrebbero escluse: applicando il calcolo dell'indice col pacchetto `statsmodels.stats.outliers_influence` di Python sono state trovate alcune variabili con valori dell'indice superiori alla soglia (*GestationalAge*, *Weight*, *C12* e *C14*, ad esempio); tuttavia, su richiesta specifica dei clinici dell'Ospedale Buzzi, non è stato possibile escludere variabili quantitative così importanti dalle analisi in corso (essendo inoltre variabili con percentuali bassissime di valori mancanti), dunque sono state mantenute all'interno del dataset per le fasi successive.

2.1.4 Variabili qualitative

Per quanto riguarda le variabili qualitative, tramite l'utilizzo del metodo `.describe()` sono state riportate le statistiche relative alle distinte categorie che compongono ogni variabile, al numero di osservazioni presenti per ogni variabile, e alla moda di ogni feature qualitativa del dataset (con la relativa frequenza).

In seguito, tramite il metodo `.value_counts()` sono state indicate tutte le occorrenze delle singole variabili qualitative con le relative frequenze per ogni categoria.

All'interno della sezione di analisi delle variabili qualitative, inoltre, sono stati inseriti i barplot [python-graph-gallery.com] relativi alle distribuzioni delle singole variabili.

2.1.5 Ulteriori esplorazioni

Grazie a delle crosstabulazioni [Banks 04] effettuate tramite la funzione `.crosstab()` del modulo *Pandas* tra differenti variabili qualitative, sono state verificate le distribuzioni congiunte tra diverse colonne del dataset, per osservare eventuali patterns di interesse. È stata utilizzato, inoltre, il pacchetto *tableOne* [pypi.org 2] per la creazione di crosstabulazioni.

Inoltre, le funzioni `.groupby()`, `.crosstab()` e `.describe()` hanno permesso di raccogliere statistiche ed informazioni sull'intero dataset stratificato in base ad alcune variabili qualitative di interesse: in questo modo è possibile vedere l'effetto di determinate categorie su alcune distribuzioni di variabili quantitative.

Infine, grazie a degli scatterplot è stato possibile indagare l'eventuale presenza/assenza di correlazione tra differenti variabili quantitative, anche in base a quanto ottenuto dall'esplorazione della matrice di correlazione e dall'heatmap corrispondente.

2.1.6 Inferenza

È stata effettuata una serie di test parametrici del tipo *t-test* (dalla libreria *scipy* di *Python*) tra gruppi di due variabili, in modo da poter verificare se le due distribuzioni delle variabili possono considerarsi significativamente differenti (ipotesi nulla è che le medie delle due distribuzioni siano uguali, l'ipotesi alternativa consiste in medie non uguali). È stato applicato anche un altro test statistico, il *Kolmogorov-Smirnov* (anche questo presente nella libreria *scipy* di *Python*): si tratta di un test statistico simile al *t-test* che permette di stabilire il grado di somiglianza di due distribuzioni, e rispetto al t-test, può essere utile per individuare differenze all'interno di distribuzioni con medie simili ma varianze molto differenti.

In caso di valori con p-value molto piccolo si possono rifiutare le ipotesi nulle per cui le medie delle due distribuzioni possono essere considerate uguali.

2.2 Risultati e discussione

2.2.1 Statistiche descrittive variabili quantitative

Come indicato in precedenza [2.1.2], nella tabella seguente vengono riportati, in ordine, l'unità di misura delle variabili, la quantità di osservazioni per cui ci sono valori non nulli, la percentuale di valori mancanti, il valore minimo, i quantili 2.5%, 25% (primo quartile), 50% (o mediana), 75% (terzo quartile) e 97.5%, il valore massimo, il range interquartile (ovvero il range che intercorre tra il valore del terzo e del primo quartile), la skewness (indice che misura l'asimmetria di una distribuzione, con valori pari a 0 per distribuzioni simmetriche,) e la curtosi (indice che fornisce indicazioni relative al grado di "appiattimento" della curva di una distribuzione, con valori superiori a 3 per distribuzioni con code "pesanti" e con valori minori di 3 per distribuzioni con code "leggere"), i valori di media, deviazione standard e infine la percentuale di outliers (considerati come i valori inferiori al 2.5% quantile e superiori al 97.5% quantile).

I risultati ottenuti per le statistiche descrittive delle variabili quantitative, effettuate sul campione di 295738 osservazioni, sono i seguenti (non vengono riportati gli intervalli di confidenza al 95%, in quanto si tratta di valori molto vicini alle statistiche stesse, nell'ordine di <0.01%, per cui l'intervallo di confidenza riportato sarebbe overkilling come informazione):

	Unità misura	Count	% NaN	Min	2.5%	25%	50% or 75% median	97.5%	Max	Iqr	Skew	Kurt	Mean	Std	% outlier	
ASATotal	µmol/L	20380	93%	0.03	0.1	0.18	0.23	0.32	0.72	2.77	0.14	4.06	28.51	0.27	0.17	5%
Ala	µmol/L	295057	0%	0.0	142.46	211.43	259.08	319.72	489.14	998529.9	108.29	89.44	10347.55	345.64	5143.8	5%
Allele 1		411	100%	2.0	9.0	31.0	31.0	55.0	194.0	224.0	24.0	3.11	10.91	46.0	37.21	5%
Arg	µmol/L	295057	0%	0.0	1.54	5.17	9.0	14.88	36.81	5352.47	9.71	96.99	11762.58	11.9	31.8	5%
Cit	µmol/L	295057	0%	0.0	6.04	10.93	14.03	17.92	29.67	6032.4	6.99	108.0	14840.54	15.42	33.2	5%
Glu	µmol/L	20380	3%	29.23	121.67	183.53	222.76	269.66	388.99	694.9	86.12	0.86	1.55	231.42	68.78	5%
Gly	µmol/L	295057	0%	0.0	170.03	304.83	388.03	483.54	756.47	926743.3	178.71	74.95	6167.81	520.44	7709.9	5%
Leu\Ile\Pro-OH	µmol/L	295056	0%	0.0	87.89	122.58	146.71	177.66	264.72	348341.4	55.08	77.71	6521.68	193.58	2778.6	5%
Orn	µmol/L	295057	0%	0.0	50.34	80.19	102.62	133.2	236.93	401343.3	53.01	132.97	21080.83	129.72	1751.2	5%
MET	µmol/L	295056	0%	0.0	10.26	15.92	19.65	24.01	36.64	9288.54	8.09	104.64	13836.15	21.2	46.26	5%
PHE	µmol/L	295060	0%	0.12	33.35	45.58	53.1	61.86	86.53	197180.6	16.28	229.83	56721.54	59.11	640.63	5%
TYR	µmol/L	295060	0%	0.0	44.54	70.54	89.07	113.41	192.97	352349.8	42.87	159.84	29517.36	109.64	1537.1	5%
HCYS	µM	1002	100%	0.0	1.03	2.1	2.76	3.79	7.03	76.3	1.69	17.77	448.4	3.17	2.83	5%
Pro	µmol/L	295057	0%	0.0	108.64	150.31	176.92	209.34	300.74	527548.2	59.03	76.82	6960.62	238.57	3575.6	5%
Val	µmol/L	295057	0%	0.0	74.15	109.1	131.88	159.41	233.29	760742.1	50.31	118.55	19647.44	178.69	3217.1	5%
BTD	U/dl	96058	68%	11.01	130.51	213.75	262.77	303.03	355.97	486.7	89.28	-0.37	-0.38	256.31	60.75	5%
C0	µmol/L	295057	0%	0.0	8.18	13.94	18.62	24.62	42.44	7978.04	10.68	85.04	10328.36	20.87	39.29	5%
C3	µmol/L	295057	0%	0.0	0.43	1.11	1.69	2.34	4.27	77.52	1.23	8.47	395.37	1.84	1.12	5%
C4OHIC3D-C	µmol/L	295057	0%	0.0	0.04	0.1	0.16	0.22	0.39	3.63	0.12	1.37	11.05	0.17	0.09	5%
C4	µmol/L	295057	0%	0.0	0.08	0.15	0.2	0.28	0.56	11.98	0.13	10.54	648.72	0.23	0.13	4%
C5OHC4D-C	µmol/L	295057	0%	0.0	0.09	0.15	0.19	0.24	0.36	26.76	0.09	85.86	19994.61	0.2	0.1	4%
C5	µmol/L	295057	0%	0.0	0.05	0.09	0.11	0.15	0.3	10.92	0.06	27.87	3159.47	0.13	0.08	3%
C5:1	µmol/L	295057	0%	0.0	0.0	0.01	0.01	0.01	0.02	0.4	0.0	5.78	149.02	0.01	0.01	1%
C5DC\C6O-H	µmol/L	295057	0%	0.0	0.05	0.08	0.11	0.14	0.23	30.29	0.06	195.45	66371.82	0.12	0.08	5%
C6	µmol/L	295057	0%	0.0	0.02	0.03	0.04	0.05	0.09	3.28	0.02	18.77	2099.25	0.04	0.02	3%
C6DC	µmol/L	295057	0%	0.0	0.04	0.08	0.11	0.15	0.24	15.53	0.07	60.24	15212.98	0.12	0.06	4%
C8	µmol/L	295057	0%	0.0	0.02	0.04	0.05	0.07	0.13	37.58	0.03	279.79	93362.2	0.06	0.1	3%
C8:1	µmol/L	295057	0%	0.0	0.01	0.02	0.03	0.05	0.18	2.65	0.03	5.69	140.23	0.05	0.05	3%
C10	µmol/L	295057	0%	0.0	0.03	0.06	0.08	0.11	0.21	2.51	0.05	3.84	69.06	0.09	0.05	4%
C10:1	µmol/L	295057	0%	0.0	0.02	0.04	0.05	0.06	0.09	1.12	0.02	4.93	116.18	0.05	0.02	3%
C10:2	µmol/L	295057	0%	0.0	0.0	0.0	0.0	0.01	0.02	7.66	0.01	425.43	212345.3	0.0	0.02	2%
C12	µmol/L	295057	0%	0.0	0.03	0.06	0.09	0.13	0.28	11.18	0.07	15.11	2047.27	0.11	0.07	4%
C12:1	µmol/L	295057	0%	0.0	0.02	0.04	0.06	0.1	0.22	1.5	0.06	2.24	12.21	0.08	0.06	5%
C14	µmol/L	295057	0%	0.0	0.06	0.14	0.2	0.26	0.43	7.24	0.12	3.31	143.45	0.21	0.1	5%

C14:1	μmol/L	295057	0%	0.0	0.02	0.06	0.1	0.14	0.29	11.42	0.08	14.4	1691.41	0.11	0.08	4%
C14:2	μmol/L	295057	0%	0.0	0.01	0.01	0.02	0.02	0.04	0.71	0.01	7.42	334.34	0.02	0.01	3%
C14-OH	μmol/L	295057	0%	0.0	0.0	0.01	0.01	0.02	0.04	0.45	0.01	2.2	31.3	0.02	0.01	1%
C16	μmol/L	295057	0%	0.0	0.49	2.0	3.1	4.07	6.31	75.53	2.07	3.33	93.42	3.08	1.66	5%
C16:1	μmol/L	295057	0%	0.0	0.02	0.12	0.21	0.28	0.45	11.03	0.16	2.99	229.8	0.21	0.12	3%
C16-OH	μmol/L	295057	0%	0.0	0.01	0.01	0.02	0.03	0.05	2.34	0.02	55.28	8391.13	0.02	0.01	3%
C16:1-OH	μmol/L	295057	0%	0.0	0.01	0.03	0.04	0.05	0.08	0.86	0.02	1.56	35.15	0.04	0.02	2%
C18	μmol/L	295057	0%	0.0	0.26	0.67	0.93	1.2	1.88	32.85	0.53	7.53	321.27	0.96	0.46	5%
C18:1	μmol/L	295057	0%	0.0	0.54	1.14	1.51	1.92	2.9	43.54	0.78	9.16	391.07	1.56	0.68	5%
C18:2	μmol/L	295057	0%	0.0	0.06	0.12	0.16	0.23	0.49	11.14	0.11	9.85	610.04	0.19	0.12	5%
C18-OH	μmol/L	295057	0%	0.0	0.0	0.01	0.01	0.02	0.03	3.38	0.01	121.39	36883.77	0.01	0.01	1%
C18:1-OH	μmol/L	295057	0%	0.0	0.01	0.02	0.02	0.03	0.04	1.59	0.01	25.7	2769.31	0.02	0.01	3%
C18:2OH	μmol/L	20380	93%	0.0	0.01	0.01	0.01	0.01	0.02	0.08	0.0	2.48	13.6	0.01	0.0	4%
C20	μmol/L	20380	93%	0.0	0.01	0.02	0.04	0.05	0.11	0.5	0.03	3.77	29.16	0.04	0.03	4%
C22	μmol/L	20380	93%	0.0	0.01	0.01	0.01	0.01	0.03	0.11	0.0	4.02	26.19	0.01	0.01	4%
C24	μmol/L	20380	93%	0.0	0.01	0.01	0.02	0.02	0.04	0.16	0.01	3.22	19.65	0.02	0.01	4%
C26	μmol/L	203800	93%	0.0	0.0	0.01	0.01	0.01	0.03	0.16	0.01	4.82	40.63	0.01	0.01	3%
SA	μmol/L	295060	0%	0.0	0.22	0.57	0.78	1.02	1.51	21.61	0.45	7.93	281.51	0.8	0.39	5%
ADO	μmol/L	20380	93%	0.08	0.27	0.44	0.57	0.73	1.31	8.39	0.29	5.22	65.28	0.63	0.32	5%
D-ADO	μmol/L	20380	93%	0.0	0.01	0.01	0.02	0.02	0.06	0.36	0.01	7.53	80.44	0.02	0.02	4%
C20:0-LPC	μmol/L	20380	93%	0.02	0.11	0.19	0.29	0.48	1.23	5.2	0.29	2.93	16.02	0.39	0.31	5%
C22:0-LPC	μmol/L	20380	93%	0.02	0.1	0.16	0.21	0.29	0.73	3.6	0.13	3.54	24.62	0.26	0.17	5%
C24:0-LPC	μmol/L	20380	93%	0.04	0.22	0.36	0.45	0.58	1.25	3.12	0.23	2.49	9.64	0.51	0.26	5%
C26:0-LPC	μmol/L	20380	93%	0.03	0.12	0.2	0.25	0.33	0.78	2.62	0.13	3.71	21.59	0.3	0.18	5%
s-TSH	μU/mL	13	100%	2.96	4.58	8.95	24.1	33.08	475.0	619.0	24.13	3.36	11.59	74.18	167.35	15%
IRT-GSP	ng/mL	352	100%	0.18	7.81	12.36	16.51	22.3	42.62	115.4	9.94	3.25	21.07	19.07	10.66	5%
TGAL	mg/dl	95982	68%	0.0	0.0	0.96	1.72	2.83	6.4	470.46	1.87	93.26	19111.92	2.12	2.26	3%
s-17OHP		1	100%	487.0	487.0	487.0	487.0	487.0	487.0	487.0	0.0		487.0		0%	
MMA	μM	277262	6%	0.0	0.0	0.0	0.0	0.0	0.0	37.19	0.0	247.03	85396.04	0.0	0.1	0%
EMA	μM	1002	100%	0.0	0.29	0.63	0.87	1.18	4.0	15.69	0.55	6.14	58.47	1.08	1.03	5%
GA	μM	1002	100%	0.0	0.42	1.14	1.55	2.07	3.76	10.74	0.93	2.95	20.68	1.69	0.92	5%
2OH GA	μM	1002	100%	0.0	3.55	9.57	14.27	21.48	35.7	62.87	11.91	1.06	1.63	16.09	8.81	5%
3OH GA	μM	1002	100%	0.0	0.05	0.15	0.3	0.52	0.94	3.67	0.37	3.36	27.74	0.36	0.29	5%
3OH PA	μM	1002	100%	0.2	2.37	5.24	15.26	22.18	30.43	57.28	16.94	0.33	-0.16	14.56	8.92	5%
MCA	μM	1002	100%	0.0	0.22	4787.1	9227.0	13935.7	26286.	115527.8	9148.6	2.77	30.65	9740.8	7953.6	5%
OROTICO	μM	1002	100%	0.0	0.07	0.23	0.32	0.45	0.86	24.65	0.22	16.41	288.07	0.42	1.17	5%
PIVA	μM	1002	100%	0.0	0.0	0.0	0.0	0.0	0.05	0.99	0.0	13.8	253.92	0.01	0.04	3%
2MBC	μM	1002	100%	0.0	0.03	0.06	0.07	0.1	0.27	5.22	0.05	23.52	659.69	0.1	0.18	5%
c4-b	μM	1002	100%	0.0	0.03	0.06	0.09	0.16	0.81	1.51	0.1	3.02	10.42	0.17	0.21	5%
c4-i	μM	1002	100%	0.0	0.02	0.07	0.11	0.16	0.42	13.3	0.09	28.52	869.49	0.15	0.43	5%
Gestational Age	settim	295737	0%	23.0	32.0	38.0	39.0	40.0	41.0	43.0	2.0	-2.19	7.65	38.5	2.23	2%
weight	g	295737	0%	350.0	1665.0	2870.0	3220.0	3530.0	4130.0	5000.0	660.0	-0.94	1.96	3152.7	592.38	5%

Table 2.1: tabella informazioni e statistiche descrittive variabili quantitative

È possibile osservare che all'interno del dataset è presente un numero considerevole di variabili con un'alta percentuale di valori mancanti: in ordine, *s-17OHP* ha solo un'osservazione (di cui non è riportata l'unità di misura), *s-TSH* (che indica la quantità di un ormone che stimola la tiroide) solo 13, *IRT-GSP* (indica la quantità di tripsin(ogeno) reattivo) solamente 352, *Allele 1* (indica l'eventuale presenza di una variante genetica associata a malattie ereditarie, e di cui non è riportata l'unità di misura) ha solamente 441 osservazioni.

Le variabili *HCVS* (omocisteina), *EMA* (acido etilmalonico), *GA* (acido glutarico), *2OH GA* (acido 2-OH glutarico), *3OH GA* (acido 3-OH glutarico), *3OH PA* (acido 3-OH propionico), *MCA* (acido metilcitrico), *OROTICO* (acido orotico), *PIVA* (pivaloilcarnitina), *2MBC* (2-metilbutirrilcarnitina), *c4-b* (butirrilcarnitina) e *c4-i* (isobutirrilcarnitina) hanno tutte 1002 osservazioni, dunque possono indicare dei soggetti con caratteristiche particolari (o appartenenti allo stesso ospedale che ha apparecchiature differenti e dunque ha rilevato ulteriori caratteristiche, oppure soggetti che fanno parte di un cluster con caratteristiche differenti dal resto dei neonati); il tutto andrebbe indagato ulteriormente.

Altre 13 variabili hanno circa il 93% di valori totali mancanti sul totale, dunque è importante prestare particolare attenzione a questi valori mancanti, che possono portare ad algoritmi con risultati poco validi e molto lontani dalla realtà dei fatti (oltre al fatto che la presenza di outliers nelle distribuzioni di queste variabili possono influire sensibilmente su alcuni indicatori statistici, come la media).

Tra le features più importanti abbiamo l'età gestazionale (definita dalla variabile *GestationalAge*, ovvero il numero di settimane intercorse tra l'ultimo ciclo mestruale prima del concepimento e il giorno del parto), con valore medio di 38.5 settimane e deviazione standard di 2.23 settimane: la distribuzione è fortemente asimmetrica verso destra, con il 75% delle madri con tempi sopra le 38 settimane (al massimo 43 settimane), ed un minimo di 23 settimane.

Un'altra variabile fondamentale è la variabile *Weight*, che indica il peso dei neonati subito dopo il parto: con media pari a 3152.74g e deviazione standard pari a 592.38g, gran parte della distribuzione (dal primo al terzo quartile) è racchiusa in un intervallo di valori abbastanza circoscritto (da 2870g a 3530g), mentre i valori estremi sono abbastanza distanti da questi valori (350g come valore minimo, 5000g come valore massimo), sinonimo di parti o gravidanze con alcune problematiche.

Una serie di variabili presentano outliers molto distanti dalla distribuzione dei dati: questi possono condizionare fortemente i valori di media e deviazione standard delle distribuzioni: ad esempio *Ala* (ovvero l'alanina), con media pari a 345.64 $\mu\text{mol/L}$ e deviazione standard di 5143.85 $\mu\text{mol/L}$, ha un valore massimo di 998529.91 $\mu\text{mol/L}$, e allo stesso modo la variabile *Arg* (l'arginina, con media pari a 11.9 $\mu\text{mol/L}$ e deviazione standard di 31.8 $\mu\text{mol/L}$, ma valore massimo di 5352.47 $\mu\text{mol/L}$), la variabile *Cit* (citrullina, con media pari a 15.42 $\mu\text{mol/L}$ e deviazione standard di 33.20 $\mu\text{mol/L}$, ma valore massimo di 6032.40 $\mu\text{mol/L}$), la variabile *Gly* (glicina, media pari a 520.44 $\mu\text{mol/L}$ e deviazione standard di 7709.97 $\mu\text{mol/L}$, ma valore massimo di 926743.38 $\mu\text{mol/L}$), la variabile *Leu\Ile\Pro-OH* (leucina/soleucina/idrossiprolina, con media pari a 193.58 $\mu\text{mol/L}$ e deviazione standard di 2778.62 $\mu\text{mol/L}$, ma valore massimo di 348341.42 $\mu\text{mol/L}$), la variabile *Orn* (ornitina, con media pari a 129.72 $\mu\text{mol/L}$ e deviazione standard di 1751.22 $\mu\text{mol/L}$, ma valore massimo di 401343.34 $\mu\text{mol/L}$), la variabile *MET* (metionina, media pari a 21.20 $\mu\text{mol/L}$ e deviazione standard di 46.26 $\mu\text{mol/L}$, ma valore massimo di 9288.54 $\mu\text{mol/L}$), la variabile *PHE* (fenilalanina, con media pari a 59.11 $\mu\text{mol/L}$ e deviazione standard di 640.63 $\mu\text{mol/L}$, ma valore massimo di 197180.64 $\mu\text{mol/L}$), la variabile *TYR* (tirosina, con media pari a 109.64 $\mu\text{mol/L}$ e deviazione standard di 1537.12 $\mu\text{mol/L}$, ma valore massimo di 352349.87 $\mu\text{mol/L}$), la variabile *Pro* (prolina, con media pari a 238.57 $\mu\text{mol/L}$ e deviazione standard di 3575.64 $\mu\text{mol/L}$, ma valore massimo di 527548.27 $\mu\text{mol/L}$) e la variabile *Val* (valina, con media pari a 178.69 $\mu\text{mol/L}$ e deviazione standard di 3217.18 $\mu\text{mol/L}$, ma valore massimo di 760742.12 $\mu\text{mol/L}$). Andrebbero osservate maggiormente nello specifico le distribuzioni, per verificare se si tratta di una sola osservazione (o solamente alcune) molto lontana dagli altri valori assunti dalla variabile, o se si tratta di una serie di outliers con valori molto distanti dal resto della distribuzione da indagare ulteriormente.

2.2.2 Grafici variabili quantitative

Tramite i packages *seaborn* e *matplotlib* di *Python* sono stati creati una serie di istogrammi relativi alle variabili quantitative presenti nel dataset (non sono stati riportati gli istogrammi delle variabili *s-17OHP* e *s-TSH* in quanto si tratta di due variabili con rispettivamente 1 e 13 osservazioni), senza gli outliers (definiti come i valori inferiori al quantile 2.5% e superiori al quantile 97.5%).

Non essendo il focus principale di questo progetto di tesi, i grafici sono stati riportati nell'Appendice 1 [Figure analisi esplorative].

Variabili come *C12* (dodecanoilcarnitina, con unità di misura $\mu\text{mol/L}$) e *Ala* (alanina, con unità di misura $\mu\text{mol/L}$) seguono l'andamento generale del dataset, con predominanza di distribuzioni asimmetriche negative, ovvero che presentano gran parte delle osservazioni in corrispondenza di valori molto bassi: quasi tutte le distribuzioni delle variabili sono di questo tipo (alcune presentano una variabilità superiore, ma comunque abbastanza limitata, come mostrato da histogrammi asimmetrici con coda allungata verso destra), ad eccezione di alcune variabili aventi una percentuale di valori mancanti elevata, oppure altre variabili come *GestationalAge* e *Weight*, più asimmetriche positive ma in maniera meno evidente delle altre variabili.

Per rendere le visualizzazioni maggiormente leggibili, sono state riportate le distribuzioni delle variabili escludendo una serie di outliers particolarmente elevati rispetto al resto delle distribuzioni (ad esempio, solo 4 osservazioni per *C12*, tutte superiori al valore 2, non vengono mostrate nell'istogramma, un numero veramente esiguo rispetto alle 295053 rimanenti; allo stesso modo, 66 osservazioni sono state escluse per la variabile *Ala*, tutte superiori al valore 100000, e mantenendo le restanti 294991).

2.2.3 Analisi correlazioni

È stato possibile ottenere la matrice di correlazione, e la relativa heatmap, del dataset:

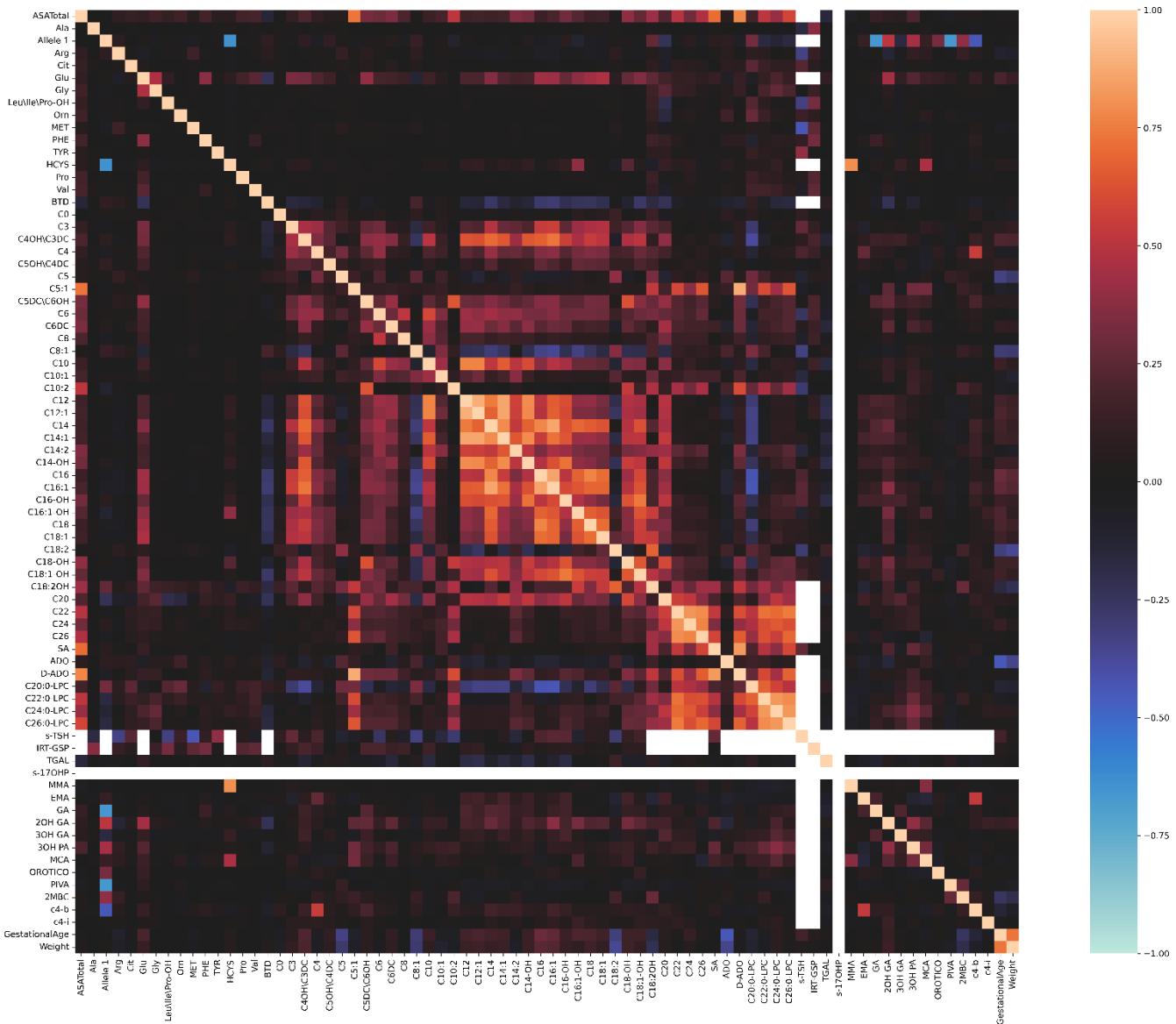


Figure 2.2: heatmap matrice di correlazione completa (no valori)

Le celle bianche della heatmap [2.2] sono relative alle variabili con pochissimi valori presenti (dunque le variabili *s-17OHP*, *s-TSH* e *IRT-GSP*, che contano rispettivamente 1, 13 e 352 osservazioni, come osservato nella sezione relativa alle statistiche descrittive), per questo motivo i valori delle correlazioni non vengono riportati dalla matrice di correlazione. In generale è possibile notare una maggioranza di valori vicini allo 0 (ovvero tutte le celle della heatmap con colore tendente al nero), sinonimo di correlazione bassa o pressoché assente tra le variabili, e dunque di variabili incorrelate.

Tra le correlazioni con valori più elevati, notiamo che sono in gran parte correlazioni positive (colore tendente al rosso/arancio al tendere dei valori a +1), mentre una piccolissima percentuale di coppie di features ha valori negativi di correlazione (colore tendente al blu/azzurro al tendere dei valori a -1).

Osserviamo inoltre la heatmap ottenuta sulle variabili con percentuali basse di valori mancanti (sotto il 10%):

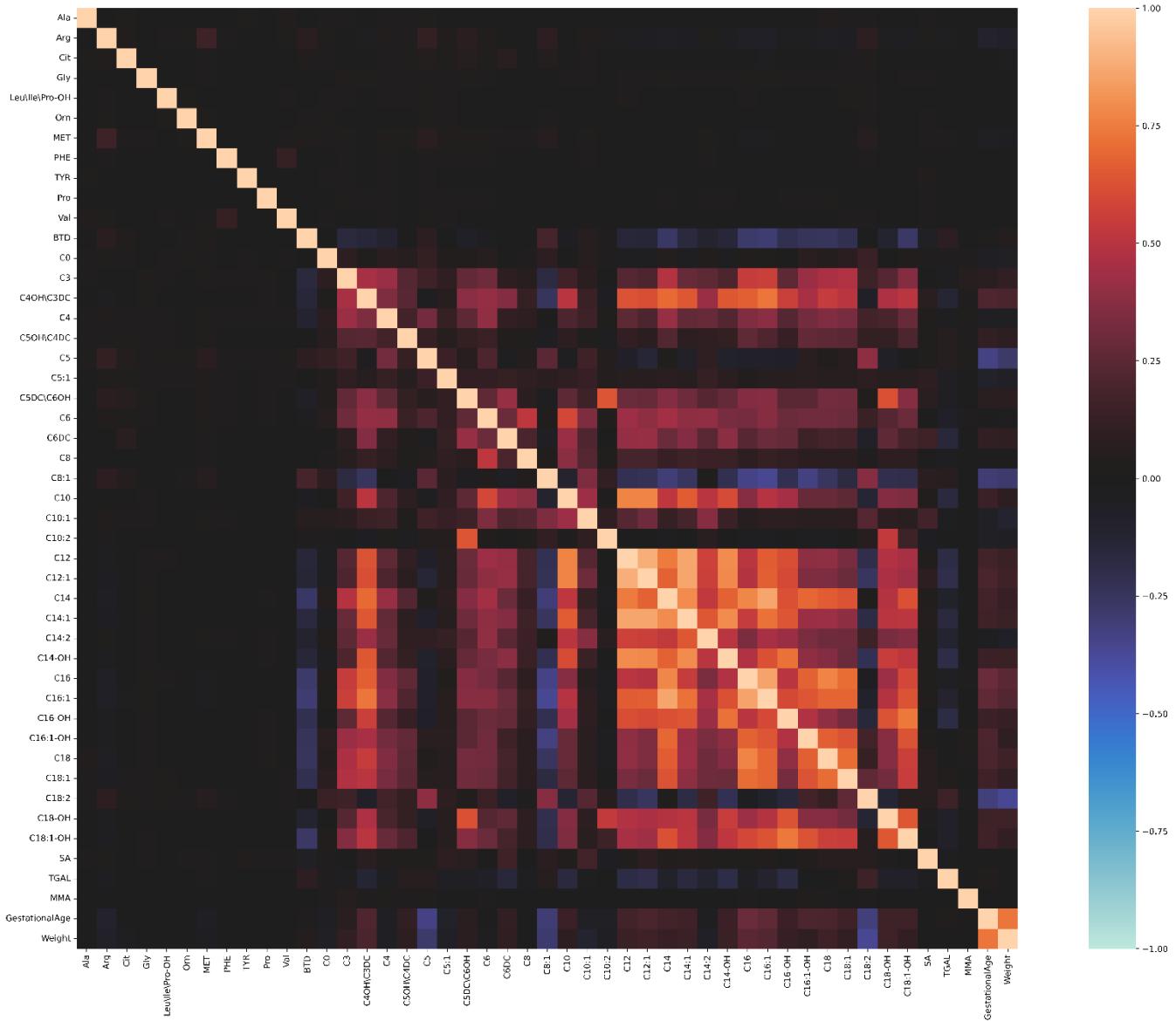


Figure 2.3: heatmap matrice di correlazione variabili con pochi dati mancanti (no valori)

Con l'accorgimento dei valori mancanti, ora è possibile notare nel grafico [2.3] ovviamente la scomparsa delle celle vuote presenti nella precedente heatmap; inoltre, come nel caso precedente, moltissime coppie di features hanno valori della correlazione tendenti allo 0, con pochi valori di correlazione positiva (i colori “caldi”) e pochissime coppie di variabili correlate negativamente (colori “freddi”).

È stato possibile osservare maggiormente nello specifico le coppie ordinate di features con valori particolarmente elevati delle correlazioni ($>|0.5|$):

Variable 1	Variable 2	Correlation
C12	C12:1	0.888
C5:1	D-ADO	0.881
C12:1	C14:1	0.865
C14	C16:1	0.865
C12	C14:1	0.86
C16	C16:1	0.859
SA	D-ADO	0.828
C24:0-LPC	C26:0-LPC	0.825
C22:0-LPC	C24:0-LPC	0.821

C22	C24	0.817
HCYS	MMA	0.799
C12	C14-OH	0.796
C14	C14:1	0.796
C14:1	C14-OH	0.793
C22:0-LPC	C26:0-LPC	0.792
C22	C26	0.791
C12:1	C14-OH	0.789
ASATotal	D-ADO	0.78
C14	C16	0.778
C10	C12:1	0.773
C10	C12	0.77
C24	C26	0.77
C16	C18	0.767
C22	C26:0-LPC	0.746
C12	C14	0.744
C16-OH	C18:1-OH	0.736
C4OHC3DC	C16:1	0.732
GestationalAge	Weight	0.729
ASATotal	C5:1	0.728
C22	C22:0-LPC	0.727
C14:1	C16:1	0.726
C14-OH	C16-OH	0.724
C18	C18:1	0.71
ASATotal	SA	0.705
C16	C18:1	0.705
C4OHC3DC	C14	0.702
C22	C24:0-LPC	0.701
C26	C26:0-LPC	0.69
C14	C16:1-OH	0.689
C16:1	C18:1	0.683
C16:1	C18	0.681
C14	C14-OH	0.68
C18:2	C18:2OH	0.678
C20:0-LPC	C22:0-LPC	0.675
D-ADO	C26:0-LPC	0.674
C12:1	C14	0.673
C16	C16:1-OH	0.672
C4OHC3DC	C16	0.67
C12:1	C16:1	0.669
C16:1-OH	C18	0.667
C10	C14:1	0.663
C12	C16:1	0.662
C14	C18	0.66
C26	D-ADO	0.657
C24	C24:0-LPC	0.651
C14:1	C14:2	0.647
C18-OH	C18:1-OH	0.645
C4OHC3DC	C14-OH	0.644
C4OHC3DC	C14:1	0.641
C5:1	C26:0-LPC	0.64
C14:1	C16-OH	0.638
C5:1	C26	0.638
C5DC\ C6OH	C10:2	0.635
C16:1	C16:1-OH	0.633
C16:1-OH	C18:1	0.633
C16:1-OH	C18:1-OH	0.631
C16-OH	C18-OH	0.631
C4OHC3DC	C12	0.629
C14	C18:1	0.629
C16:1	C18:1-OH	0.627
C26	C22:0-LPC	0.624
C5DC\ C6OH	C18-OH	0.622
C14	C16-OH	0.621
C14-OH	C16:1	0.621
C14	C18:1-OH	0.62
C24	C22:0-LPC	0.616

C22	D-ADO	0.615
C12	C16-OH	0.614
C4OH C3DC	C12:1	0.614
C24	C26:0-LPC	0.61
C10	C14-OH	0.605
C10:2	D-ADO	0.603
C5:1	C22:0-LPC	0.603
C12:1	C16-OH	0.598
C16:1	C16-OH	0.597
D-ADO	C22:0-LPC	0.594
C16	C18:1-OH	0.593
C5:1	C22	0.592
C6	C10	0.59
SA	C26:0-LPC	0.588
C18-OH	C20	0.588
ASATotal	C26:0-LPC	0.585
C26	SA	0.579
C20:0-LPC	C24:0-LPC	0.575
C12	C14:2	0.568
C12:1	C14:2	0.561
C18	C18:1-OH	0.561
C20	C22	0.56
C3	C16:1	0.558
C4OH C3DC	C16-OH	0.556
C26	C24:0-LPC	0.555
C14-OH	C18-OH	0.553
C16:1-OH	C18:2OH	0.553
C4	c4-b	0.551
C18:1	C18:1-OH	0.55
C3	C16	0.544
C14:1	C16	0.544
C4OH C3DC	C18	0.541
C10:2	C18-OH	0.538
EMA	c4-b	0.536
C18:1	C18:2OH	0.534
C14:1	C20	0.533
C5:1	C18:2OH	0.529
C14	C14:2	0.528
C22	SA	0.526
C10	C14	0.523
C14-OH	C20	0.523
C6	C8	0.521
C14-OH	C18:1-OH	0.521
C18:2OH	D-ADO	0.52
C20	C24	0.517
ASATotal	C10:2	0.514
C20	C24:0-LPC	0.514
C16-OH	C20	0.512
C14:1	C18:1-OH	0.509
ASATotal	C22:0-LPC	0.509
Allele 1	2OH GA	0.508
C12	C16	0.508
C4OH C3DC	C18:1	0.504
C10	C14:2	0.504
C14:1	C18-OH	0.504
C16	C16-OH	0.502
C4OH C3DC	C18:1-OH	0.502
C14:2	C14-OH	0.501
C22	C20:0-LPC	0.501
C4OH C3DC	C10	0.5
Allele 1	HCYS	-0.648
Allele 1	PIVA	-0.666
Allele 1	GA	-0.67

Table 2.2: coppie di variabili quantitative con valori di correlazione (positiva o negativa) >|0.5|

Sono 139 le coppie di variabili con valori di correlazione particolarmente elevati (sopra 0.5, o con valori minori di -0.5): in particolare, il valore massimo è di 0.88 tra le variabili *C12:1* e *C12*, e in generale quasi tutte le correlazioni con valori superiori a 0.5 comprendono almeno una variabile tra gli analiti del tipo *C_* (come *C10/C12/C14/C4OH*).

È interessante notare come, tra le coppie di variabili correlate positivamente, ci siano *GestationalAge* e *Weight*, con un valore di 0.729. Si tratta di un'informazione interessante, che andrebbe indagata ulteriormente, anche se è facile immaginare che a gravidanze più brevi (dunque con valori bassi di *GestationalAge*) a causa di condizioni particolari di madri e bambini corrispondano mediamente neonati con peso minore (ovvero valori più bassi di *Weight*).

Infine, è importante notare anche le uniche tre coppie di features con valori elevati di correlazione negativa: in tutti e tre i casi è compresa la variabile *Allele 1*, con valori compresi tra un minimo di -0.67 e un massimo di -0.648.

Col calcolo del VIF (Variance Inflation Factor, ovvero il Fattore di Inflazione della Varianza) è stata testata la presenza di collinearità nei dati, per cui vengono riportati solo i dati con valori particolarmente elevati dell'indicatore:

Variabile	VIF
GestationalAge	116.30
C12	76.54
C14	65.19
Weight	57.61
C16:1	47.77
C12:1	45.81

Table 2.3: valori elevati VIF variabili quantitative

Si tratterebbe di variabili da escludere a partire da ulteriori analisi; tuttavia ci è stato richiesto dai clinici di non escludere variabili come gli alleli e le due altre variabili quantitative *GestationalAge* e *Weight* poiché sono considerate features fondamentali dagli esperti di dominio, e in modelli predittivi ed algoritmi di cluster analysis sono informazioni fondamentali che vanno considerate assolutamente.

2.2.4 Analisi variabili qualitative

Per quanto riguarda, invece, le considerazioni fattibili per le variabili qualitative presenti nel dataset, è stata riportata la seguente tabella, dove la moda indica la categoria con frequenza maggiore, la colonna “unique” riporta quante categorie sono presenti per ogni variabile, mentre la frequenza assoluta viene riportata nella colonna “freq”:

	COUNT	UNIQUE	MODE	FREQ
SAMPLEQUALITY	4	1	OK	4
TPNFEED	295737	2	0	292390
ETNICITY	295735	97	Caucasian	238090
BIRTHMETHOD	242275	3	Naturale	158722
BABYFED	123411	2	1.0	122652
TPNMCTFEED	295737	2	0	295706
CORTISONEBABY	293335	2	0.0	280708
SAMPLING	295737	4	Iniziale	288540
HUFEED	295737	2	1	193483

TPNCARNFEED	295737	2	0	295364
PREMATURE	245326	2	0.0	229642
ANTIBIOTICSMOTHER	94676	2	0.0	78482
ANTIBIOTICSBABY	295736	2	0.0	274744
SEX	295737	2	M	152374
BIS	58363	2	1.0	49297
TWINS	139817	2	0.0	129257
ENFEED	295737	2	0	295259
HOSPITAL	295737	181	F.ne Cà Granda OM Policlinico Rep. NIDO	14877
TOOYOUNG	295737	2	0.0	292480
SAMPLEBARCODEANONYMIZED	295737	295737	-1433386896307634471	1
ARTFEED	295737	2	0	267521
CITY	295225	16893	MILANO	32296
CORTISONEMOTHER	94790	2	0.0	85832
TYROIDMOTHER	293586	2	0.0	250201
MECONIUM	293634	2	0.0	293178
MIXFEED	295737	2	0	232808
ANSWERIX	295737	4	1	292247
ID	295737	277056	4177155333400741261	5
REPARTO	295737	3	Nido	225284
ETNIA	295737	7	Caucasian	244921

Table 2.4: variabili qualitative del dataset

Osservando alcune delle variabili qualitative del dataset, alcuni problemi di sintassi sono subito ben chiari: ad esempio, la variabile *Etnicity* (che indica l'origine geografica) presenta 97 categorie differenti, specificando in alcuni casi direttamente la nazionalità dei genitori mentre in altri l'etnia (sono presenti sia la categoria “Caucasian” che “Italy”, ad esempio); allo stesso modo la variabile *City* spesso presenta errori di sintassi (come città riportate col nome completo, in altri casi indicate solamente con l'acronimo della provincia, come “MI” per Milano, altre ancora con la provincia tra parentesi...). Inoltre, la variabile *SampleQuality* probabilmente verrà esclusa negli steps successivi del progetto, in quanto sono riportati valori solamente per quattro neonati, come indicato in tabella.

Può essere interessante studiare gli effetti, sulle variabili quantitative, di categorie differenti per variabili relative al parto, come il fatto che sia gemellare (variabile *Twins*), o prematuro (indicato dalla variabile *Premature*, con 0 per il parto normale e 1 per il parto prematuro), o con problematiche legate ad utilizzo di antibiotici per madri o neonati (definite da features come *AntibioticsMother* e *AntibioticsBaby*)...

Alcune variabili qualitative sono abbastanza bilanciate nelle classi (come il sesso del neonato, indicato da *Sex*, con 51.3% di maschi e 48.7% di femmine), altre hanno classi abbastanza sbilanciate tra di loro (come il parto prematuro, indicato dalla variabile *Premature* con 92.9% di parti in tempi normali e 7.1% no): questo può condizionare ulteriori studi, analisi e previsioni; tuttavia, essendo ancora in una fase esplorativa del dataset, il rischio di perdere informazioni importanti escludendo una o più variabili è elevato.

In generale, è facile notare che poche variabili qualitative hanno più di due categorie, in particolare solo 8: oltre a quelle relative ad ospedale e città (rispettivamente *Hospital* e la sua derivante *Reparto*, e *City*), solo alcune relative al parto (come *Etnicity* e la sua derivante *Etnia*, *BirthMethod*) e altre riferite alla raccolta dati del campione (come *Sampling* e *AnswerIX*, che insieme indicano la fase dell'analisi) han più di due categorie.

Come descritto in precedenza [1.1.3], le due variabili Reparto ed Etnia sono state create come ricodifica di variabili qualitative pre-esistenti all'interno del dataset: la variabile *Reparto* a partire dalla variabile *Hospital*, con 225284 campioni raccolti in reparti Nido degli ospedali, 40485 in reparto generico e 29968 campioni provenienti da reparti neo-patologici; la variabile Etnia dalla variabile *Etnicity*, con 244921 madri

di origini *Caucasian*, e poi in ordine 17642 madri di origini *Arab*, 13044 madri di origini *Asian*, 12745 madri di origini *Native Hawaiian or Other Pacific Islander*, 7191 madri di origini *Hispanic/Latino*, 183 madri di origini *Black or African American* e infine 9 madri di origini *Other*.

Trattandosi di informazioni molto importanti per conoscere non solo le caratteristiche dei dati in nostro possesso, ma anche la composizione delle categorie, sono state osservate anche le tabelle di frequenze assolute (colonna “count”) e relative (colonna “%”) delle variabili qualitative, con le differenti categorie ordinate dalla più alla meno frequente (per alcune features, come Hospital ed Etnicity, con un numero particolarmente importante di categorie, sono stati riportati solamente le informazioni per le frequenze elevate):

TPNCARNFeed	count	%
0	295364	99.87%
1	373	0.13%

TooYoung	count	%
0.0	292480	98.9%
1.0	3257	1.1%

CortisoneMother	count	%
0.0	85832	90.55%
1.0	8958	9.45%

BabyFed	count	%
1.0	122652	99.38%
0.0	759	0.62%

Sex	count	%
M	152374	51.52%
F	143363	48.48%

Twins	count	%
0.0	129257	92.45%
1.0	10560	7.55%

ARTFeed	count	%
0	267521	90.46%
1	28216	9.54%

Meconium	count	%
0.0	293178	99.84%

1.0	456	0.16%
-----	-----	-------

HUFeed	count	%
1	193483	65.42%
0	102254	34.58%
BIS	count	%
1.0	49297	84.47%
0.0	9066	15.53%

AntibioticsBaby	count	%
0.0	274744	92.9%
1.0	20992	7.1%

BirthMethod	count	%
Naturale	158722	65.51%
Cesareo	74697	30.83%
Altro	8856	3.66%

AnswerIX	count	%
1	292247	98.82%
2	3450	1.17%
3	39	0.01%
4	1	0.0%

AntibioticsMother	count	%
0.0	78482	82.9%
1.0	16194	17.1%

Etnicity	count	%
Caucasian	238090	80.51%
Medio-Orientale	17443	5.9%
Asian	12858	4.35%
Afro-caraibica	12743	4.31%
Ispanic	7191	2.43%
Default	4928	1.67%
Italy	1538	0.52%
...

MIXFeed	count	%
0	232808	78.72%
1	62929	21.28%

Sampling	count	%
Iniziale	288540	97.57%
Controllo	7012	2.37%
Basale già noto	177	0.06%
BIS	8	0.0%

ENFeed	count	%
0	295259	99.84%
1	478	0.16%

TPNFeed	count	%
0	292390	98.87%
1	3347	1.13%

TyroidMother	count	%
0.0	250201	85.22%
1.0	43385	14.78%

Premature	count	%
0.0	229642	93.61%
1.0	15684	6.39%

Reparto	count	%
Nido	225284	76.18%
Generico	40485	13.69%
Neo-patologico	29968	10.13%

Hospital	Count	%
F.ne Cà Granda OM Policlinico Rep. NIDO	14877	5.03%
Ospedale Papa Giovanni XXIII Rep.NIDO	10782	3.65%
Ospedale V. Buzzi Rep. NIDO	9078	3.07%
Ospedale Poliambulanza BS REP. NIDO	8752	2.96%
Spedali Civili di Brescia Rep. NIDO	8745	2.96%
Ospedale F. del Ponte Varese Rep. NIDO	7871	2.66%
Ospedale San Gerardo REP. NIDO	7175	2.43%
Ospedale San Raffaele Rep. NIDO	6575	2.22%
Ospedale M. Melloni Rep. NIDO	5995	2.03%
Policlinico San Matteo Rep. NIDO	5813	1.97%
Ospedale Niguarda Ca' Granda Rep. NIDO	5699	1.93%
Ospedale Sant'Anna di Como Rep.NIDO	5171	1.75%
F.ne Cà Granda OM Policlinico Rep. NEO.SOLV.	4969	1.68%
Ospedale A. Manzoni Lecco Reparto NIDO	4528	1.53%
Ospedale V. Emanuele III Carate Brianza Rep. NIDO	4329	1.46%
Ospedale Bolognini di Seriate Rep.NIDO	4137	1.4%
Ospedale di Vimercate Rep. NIDO	3959	1.34%
Ospedale San Paolo Rep. NIDO	3951	1.34%
Ospedale di Circolo Busto A. Reparto NIDO	3909	1.32%
Ospedale Carlo Poma Rep.NIDO	3883	1.31%

Ospedale S. Giuseppe Rep. NIDO	3780	1.28%
F.ne Cà Granda OM Policlinico Rep. NEO.PAT	3630	1.23%
Ospedale Maggiore di Lodi Rep. NIDO	3553	1.2%
Ospedale Civile di Desio Rep. NIDO	3301	1.12%
Ospedale Valduce Como Rep. NIDO	3182	1.08%
Ospedale di Circolo di RHO Rep. NIDO	3173	1.07%
Ospedale Magenta G.Fornaroli NIDO	3142	1.06%
Ospedale P.O.C. di Cremona Rep. NIDO	3020	1.02%
...

TPNMCTFeed	count	%
0	295706	99.99%
1	31	0.01%

CortisoneBaby	count	%
0.0	280708	95.7%
1.0	12627	4.3%

City	Count	%
MILANO	32296	10.9%
MI	9229	3.12%
MONZA	3223	1.09%
BRESCIA	3177	1.07%
COMO	2200	0.74%
BERGAMO	2092	0.70%
PAVIA	1943	0.66%
BS	1884	0.64%
VARESE	1870	0.63%
CREMONA	1810	0.61%
VIGEVANO	1701	0.58%
LEGNANO	1663	0.56%
GALLARATE	1642	0.56%
LISSONE	1603	0.55%
...

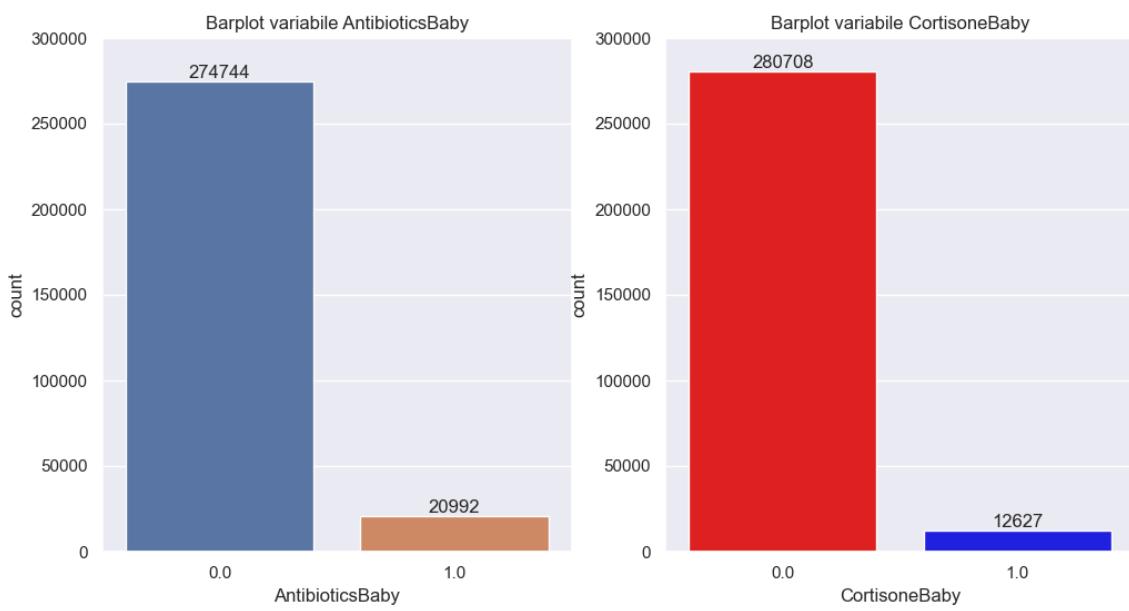
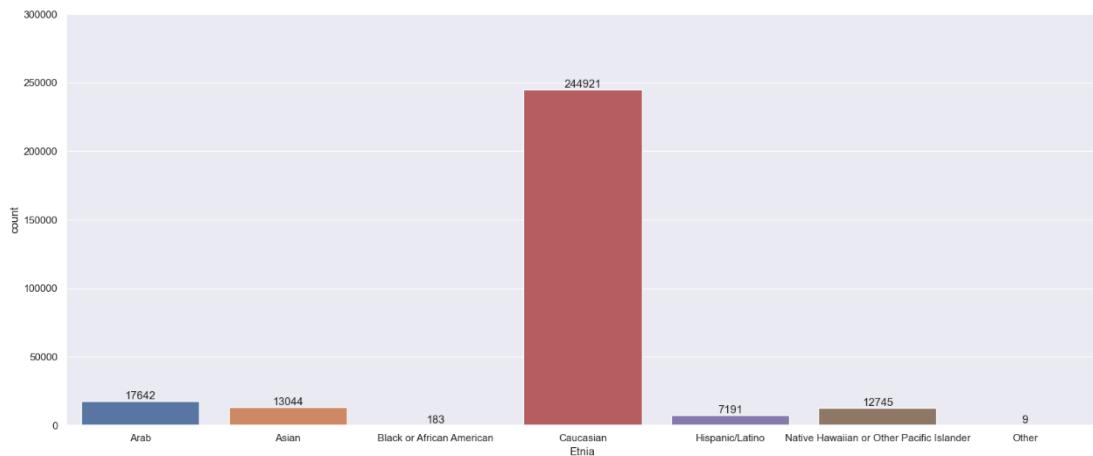
Etnia	count	%
Caucasian	244921	82.84%
Arab	17642	5.98%
Asian	13044	4.42%
Native Hawaiian or Other Pacific Islander	12745	4.32%
Hispanic/Latino	7191	2.44%
Black or African American	183	0.0%
Other	9	0.0%

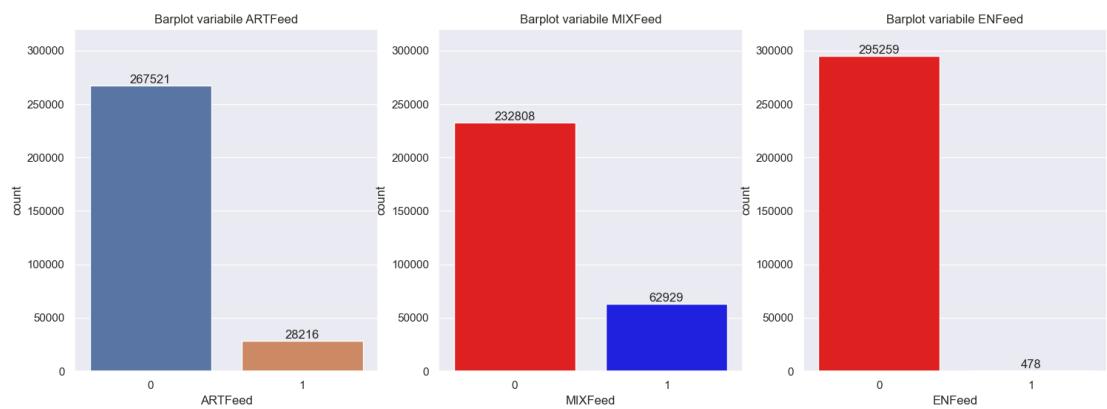
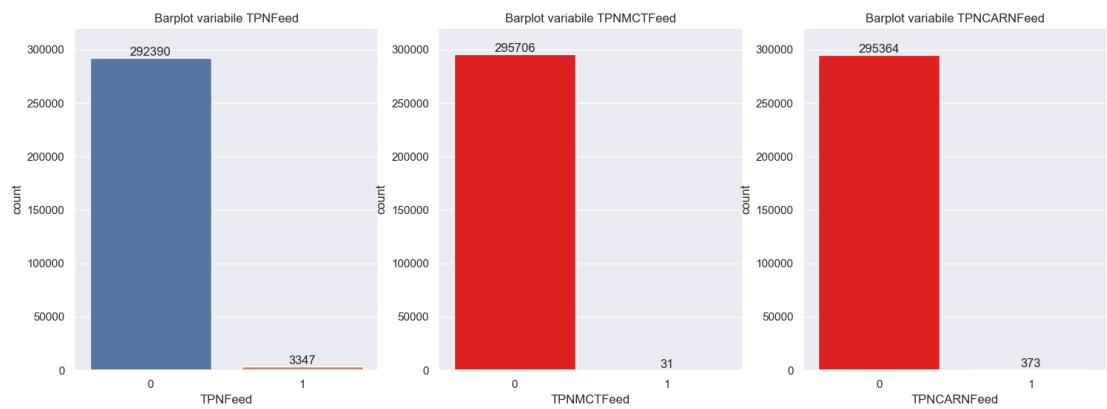
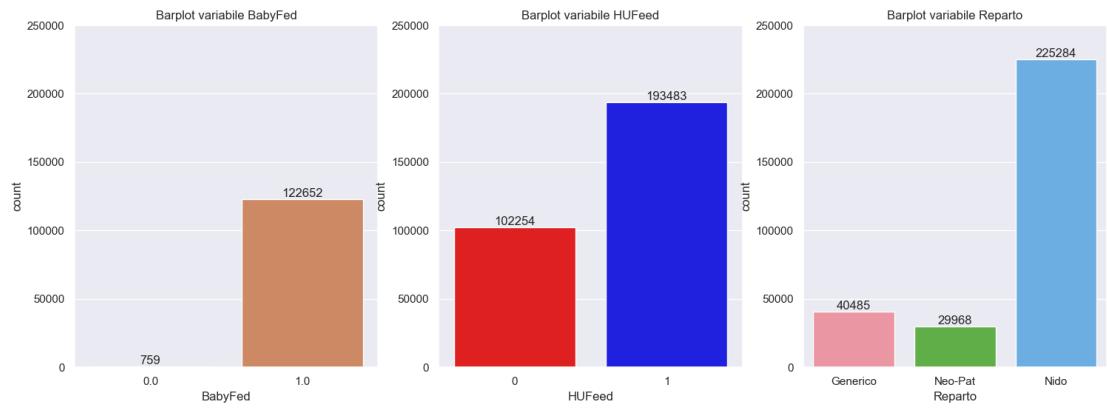
Table 2.5: tabelle distribuzioni variabili qualitative

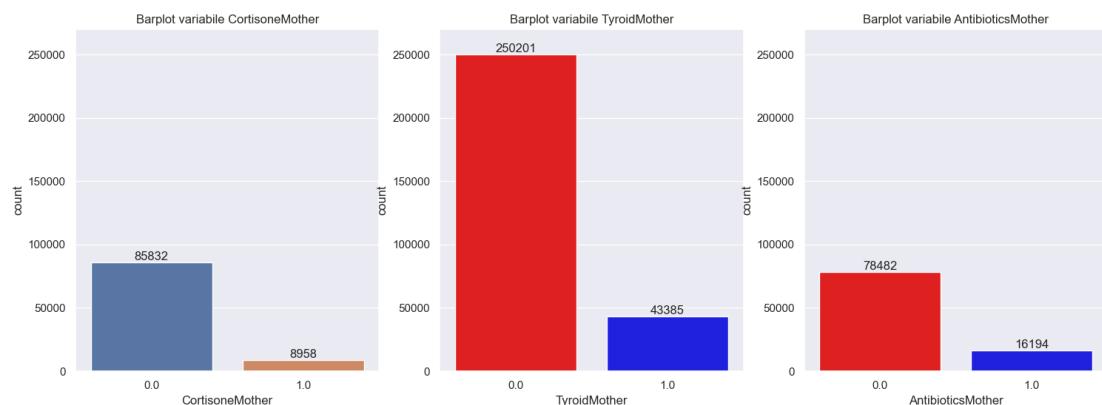
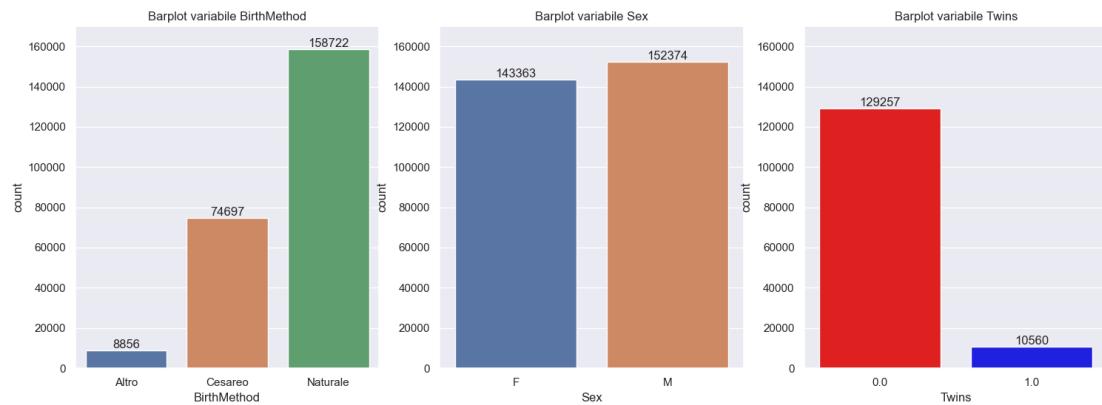
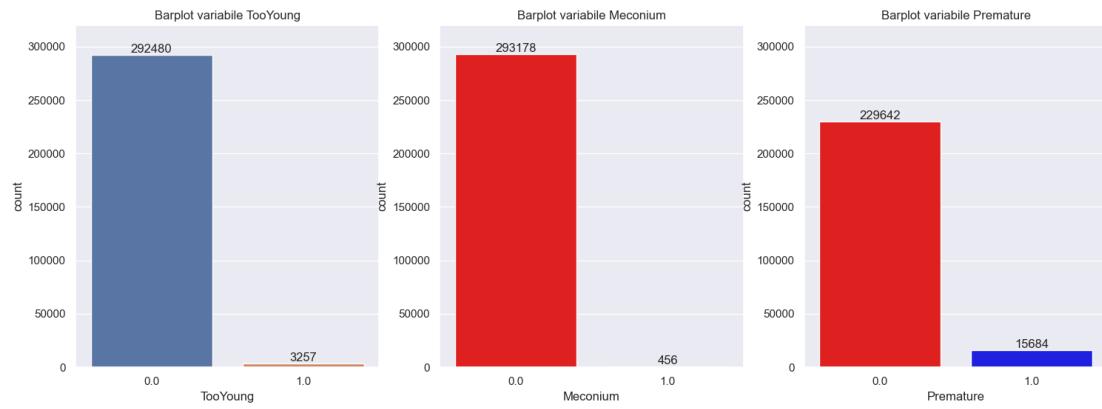
Dall'analisi più approfondita sulle variabili qualitative, è possibile osservare che la variabile *City* necessiterebbe di ulteriore attenzione e pulizia, in quanto sono presenti molti errori di sintassi e di lessico; tuttavia, si tratta comunque di un aspetto marginale rispetto agli obiettivi di questo progetto di ricerca.

2.2.5 Barplot variabili qualitative

Con la creazione di barplot è possibili analizzare ulteriormente le distribuzioni delle variabili qualitative presenti all'interno del dataset, con particolare attenzione alle differenze in ordini di grandezza tra alcune categorie particolarmente prevalenti in determinate distribuzioni. Sono state osservate graficamente solo le features con un numero di categorie minore di 10 (dunque escludendo le variabili *Hospital* e *City*), per confermare le considerazioni fatte nel paragrafo precedente [2.2.4]:







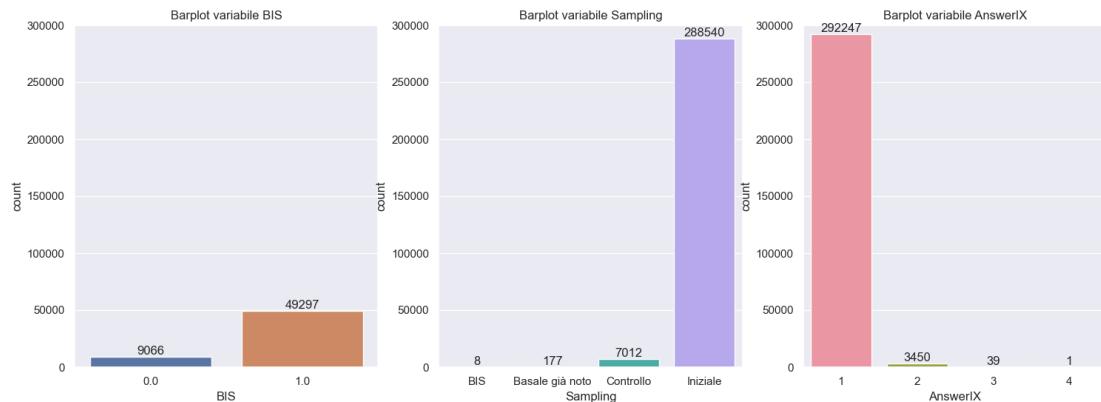
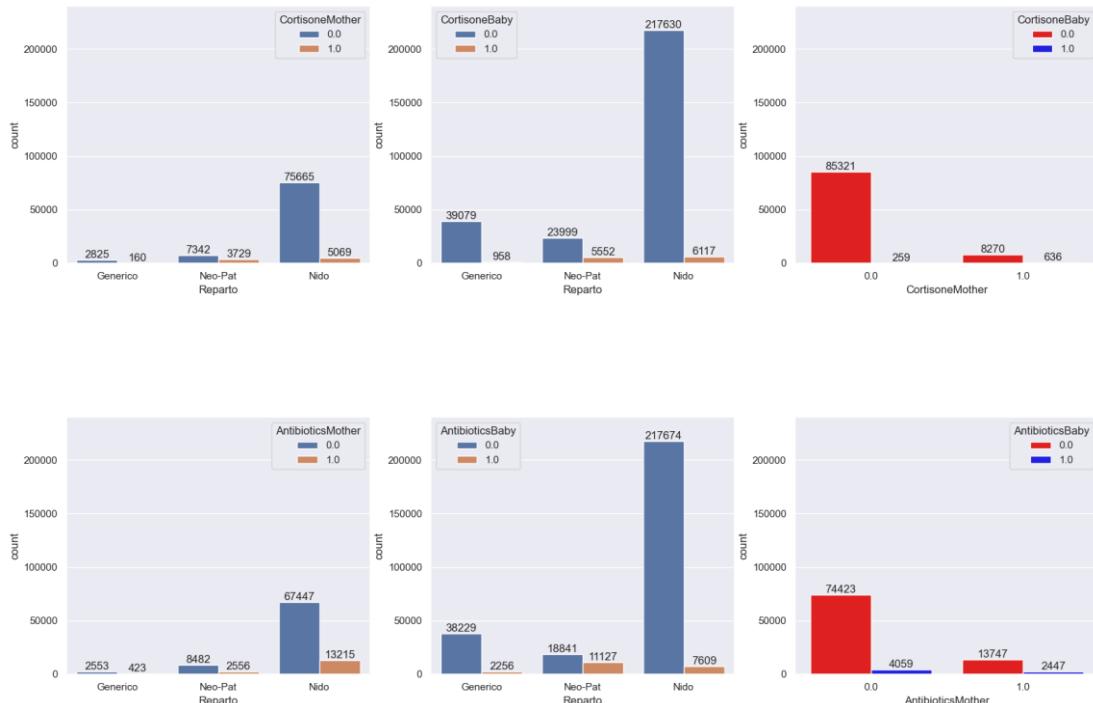


Figure 2.4: barplot variabili qualitative (escluse City ed Hospital)

Anche graficamente [2.4] vengono confermate alcune delle considerazioni fatte in precedenza [2.2.4] con le informazioni ricavate a partire dalle distribuzioni delle variabili qualitative: molte features hanno categorie di dimensioni non proporzionate, con spesso una classe predominante (con percentuali molto elevate sul totale dei dati) e le altre con poche osservazioni.

2.2.6 Esplorazioni ulteriori

In questa fase di esplorazioni dei dati a disposizione, e delle relazioni che intercorrono tra le features presenti nel dataset, possono essere fatte una serie di considerazioni andando a confrontare le distribuzioni delle variabili qualitative rispetto ad altre features, sia qualitative che quantitative.



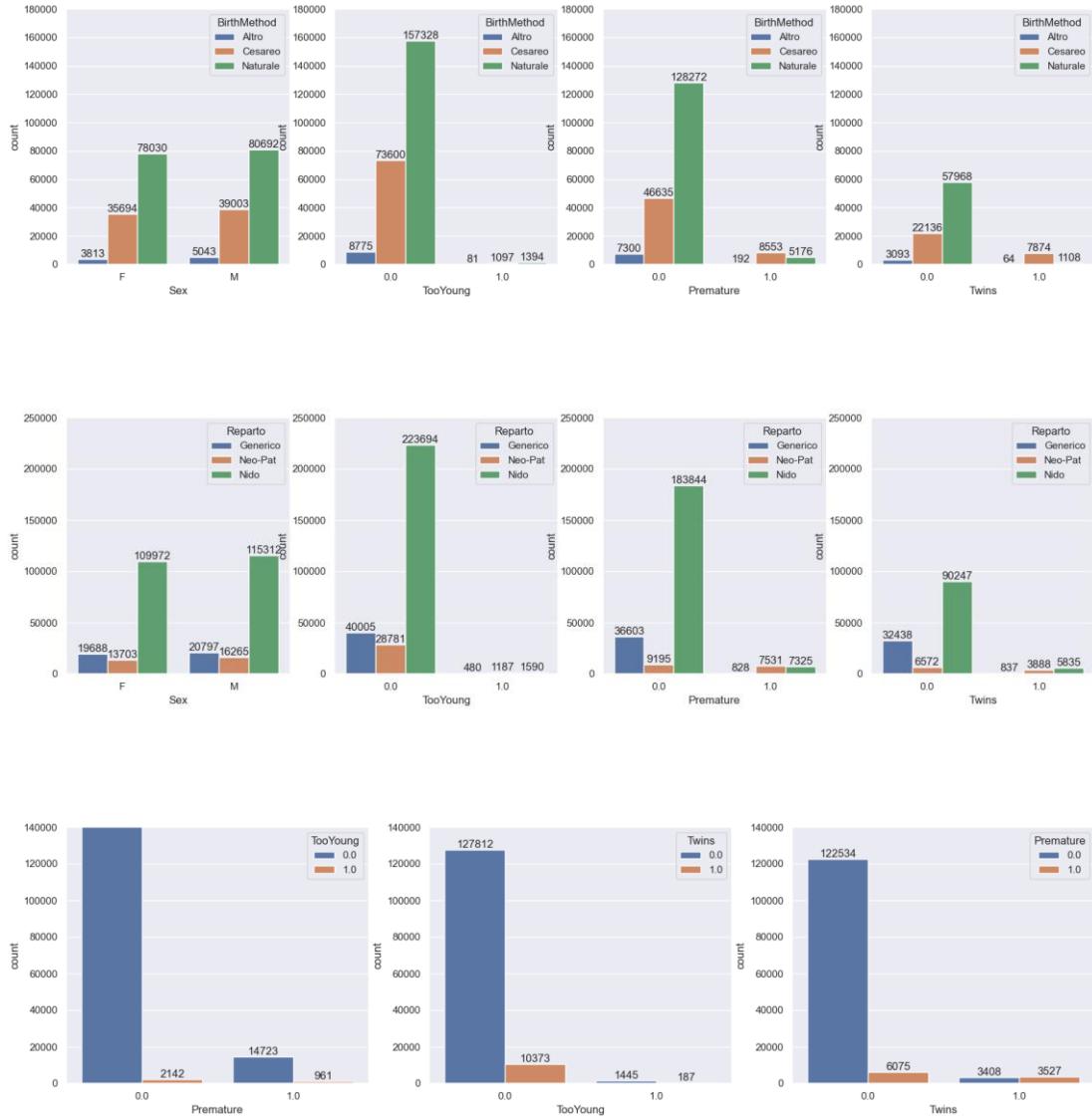


Figure 2.5: esplorazione variabili qualitative con barplot stratificati

Il confronto tra variabili differenti [2.5] permette di ottenere insights interessanti: nei confronti tra CortisoneMother, CortisoneBaby, AntibioticsMother e AntibioticsBaby con la variabile reparto, si può notare una forte differenza percentuale in caso di assunzione di antibiotici o cortisone, sia da parte dei neonati che della madre; infatti, nel reparto “Neo-Patologico”, si arriva a proporzioni 1:2 tra neonati con madri che hanno assunto cortisone, e ancora più alti per bambini a cui sono stati somministrati antibiotici, ponendo in risalto le problematicità dei bambini ricoverati in reparti neo-patologici.

Dai barplot successivi, risaltano subito alcune features che non incidono né su *BirthMethod* né su *Reparto*, come il sesso del neonato (in generale, il sesso del neonato sembra indipendente a qualsiasi condizione particolare in cui avviene il parto), ed altre che sono molto significative per le distribuzioni dei reparti e delle tipologie di parto, come i parti prematuri e i parti gemellari. Da notare, inoltre, alcuni casi particolari come i parti gemellari, che il 50% risultano come nascite premature (, mentre solamente il 5% circa dei parti non gemellari si verifica come parto prematuro).

Inoltre, altre assunzioni possono essere fatte se si confrontano distribuzioni di variabili quantitative differenziate per una feature qualitativa [2.6] (il nostro focus è stato sempre sulle variabili *Weight* e *GestationalAge*, essendo sia in range di valore circoscritti che di facile interpretazione e lettura):

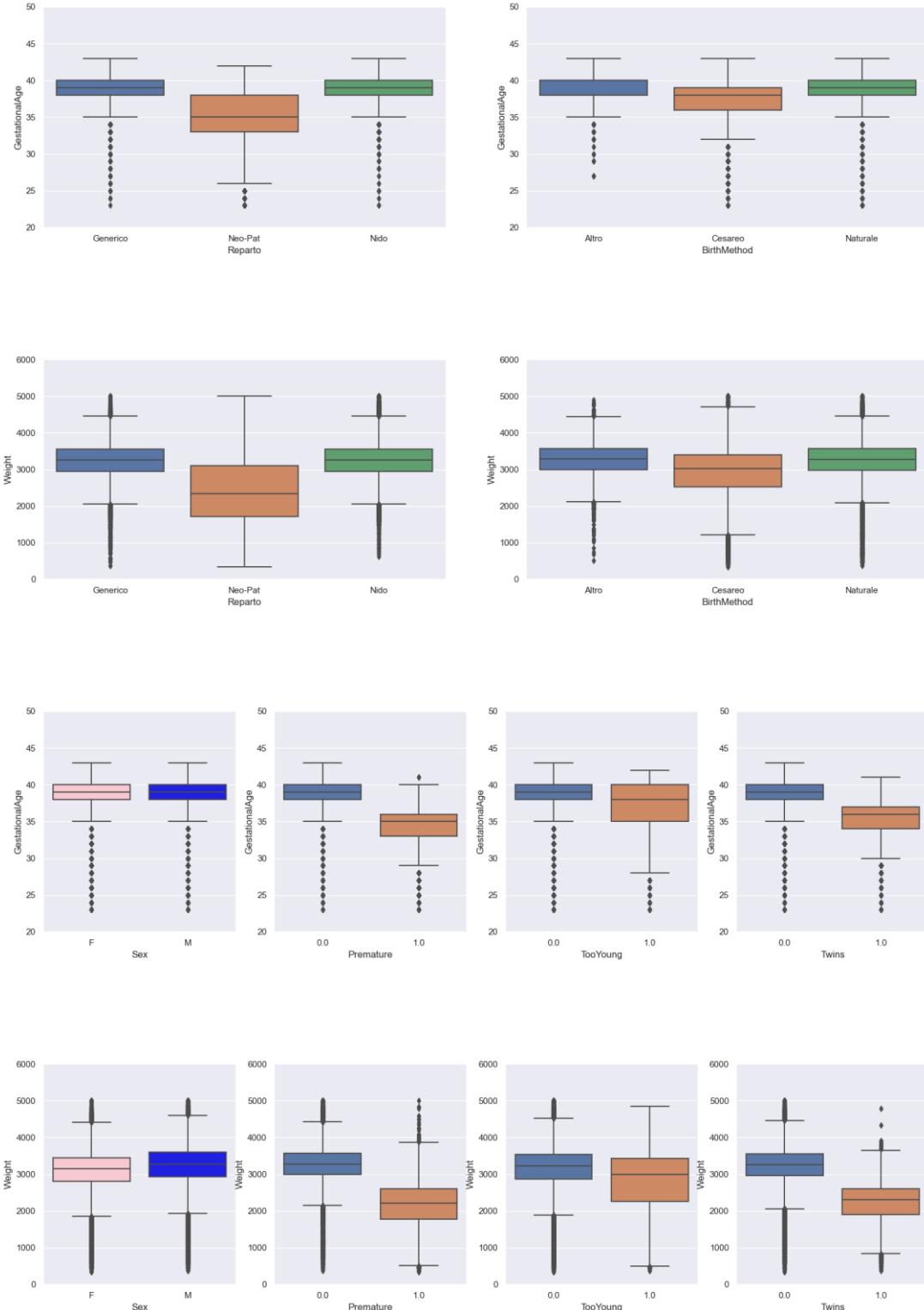


Figure 2.6: boxplot variabili Weight e Gestational Age stratificate per altre variabili qualitative

Si può notare facilmente che il sesso del neonato non condiziona né l'età gestazionale né il peso del neonato; viceversa, il fatto di avere un parto gemellare, o il fatto di avere un parto prematuro, condizionano sia il peso che l'età gestazionale, che diminuiscono mediamente in entrambi i casi; inoltre è chiara la differenza per i valori delle variabili *GestationalAge* e *Weight* in bambini ricoverati in reparto Neo-Patologico rispetto a reparto Nido e Generico, con valori molto più elevati per questi ultimi.

Infine, è stato effettuato un confronto delle distribuzioni di *Weight* e *GestationalAge* con degli scatterplot stratificati per una variabile qualitativa (ovvero le variabili *BirthMethod* e *Reparto*):

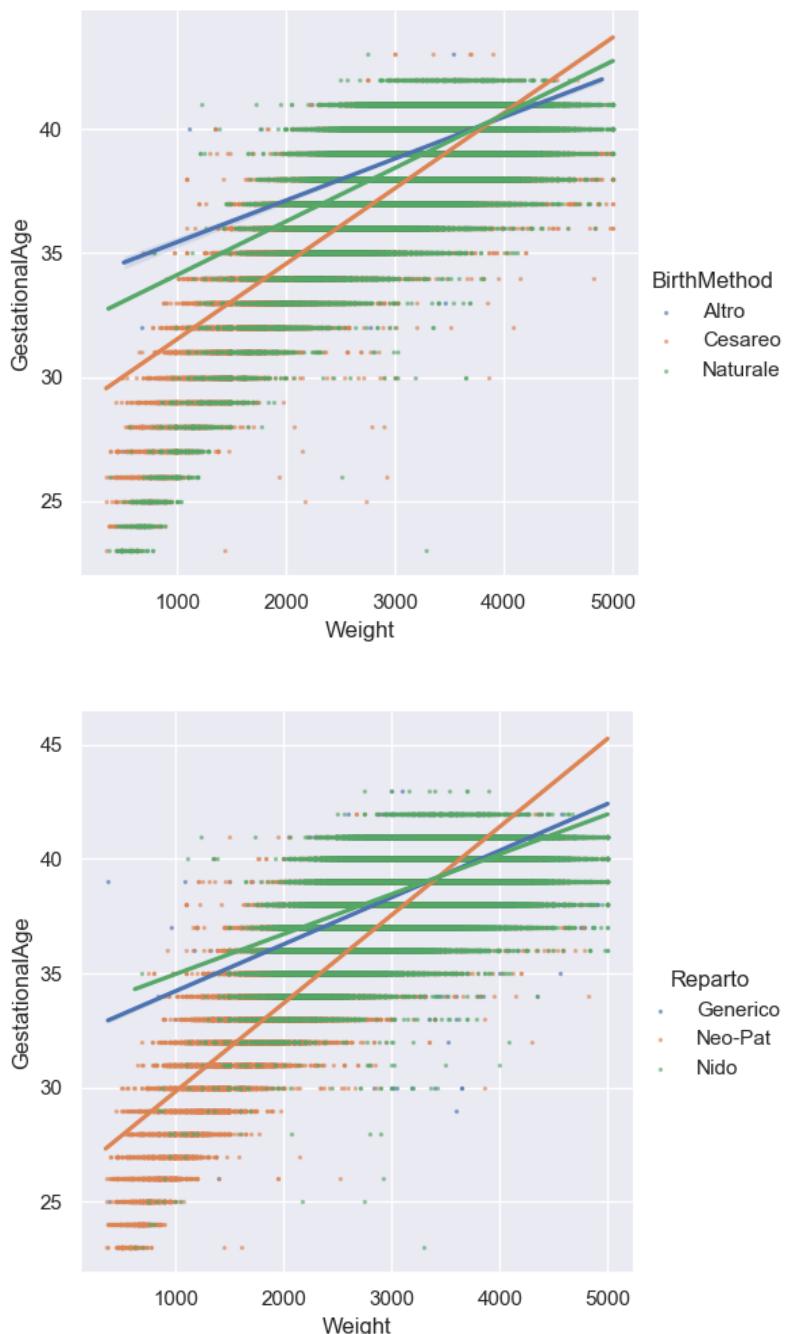


Figure 2.7: scatterplot variabili Weight e GestationalAge stratificati per variabile qualitativa

È facile notare dalla figura [2.7] che gran parte dei casi con valori bassi di *GestationalAge* e *Weight* sono tra i pazienti di un reparto neo-patologico, mentre valori elevati delle due variabili indicano solitamente appartenenza a reparto Nido o Generico.

Un comportamento simile è chiaro nel secondo scatterplot, con parti cesarei che corrispondono a valori molto piccoli delle variabili analizzate (in particolare per valori molto bassi di *Weight*)

2.2.7 Inferenza

Come descritto nel capitolo [2.1.6], per la parte di inferenza sono stati effettuati una serie di t-test tra ogni coppia di variabili continue, in modo da poter verificare se le due distribuzioni delle features d'interesse potessero considerarsi significativamente differenti (ipotesi nulla è che le medie delle due distribuzioni siano uguali, l'ipotesi alternativa consiste in medie non uguali).

Se il p-value ottenuto dal t-test è minore di 0.05, il test è considerato significativo, e dunque si rifiuta l'ipotesi nulla di uguaglianza delle medie delle due distribuzioni (e dunque di differenza delle medie delle due distribuzioni pari a 0); viceversa, se il p-value è maggiore di 0.05, non è possibile rifiutare l'ipotesi nulla di uguaglianza delle medie delle due distribuzioni. Vengono di seguito riportati i casi in cui il p-value è risultato superiore a 0.05.

	VARIABLE	T-STATISTIC	P-VALUE
73	Ala vs Allele 1	1.1809667790570861	0.23761687143371413
129	Ala vs s-TSH	0.19028078052178773	0.8490892443312431
130	Ala vs IRT-GSP	1.1911183023761713	0.23360810282337108
148	Allele 1 vs Gly	-1.2475409486481208	0.21220020414372162
149	Allele 1 vs Leu\Ile\Pro-OH	-1.0767469158309722	0.2815942256294433
150	Allele 1 vs Orn	-0.9692679115859113	0.3324123284526902
152	Allele 1 vs PHE	-0.4148345852849419	0.6782633006234698
153	Allele 1 vs TYR	-0.8393318885541264	0.4012837725207252
155	Allele 1 vs Pro	-1.0918426981043547	0.27490313810344225
156	Allele 1 vs Val	-0.8361843604556732	0.4030518715730149
346	Cit vs 2OH GA	-0.6448790124356598	0.5190060931621145
348	Cit vs 3OH PA	0.8144144815073037	0.41540820037969595
356	Glu vs Leu\Ile\Pro-OH	1.944265716776262	0.051864304407369256
362	Glu vs Pro	-0.2854645575681523	0.7752884415998876
474	Gly vs s-TSH	0.20869520975541508	0.8346863239227128
475	Gly vs IRT-GSP	1.2200453261276223	0.22244866683228448
496	Leu\Ile\Pro-OH vs Val	1.9017014737623987	0.057210679608742765
540	Leu\Ile\Pro-OH vs s-TSH	0.1549314679268239	0.8768754513245413
541	Leu\Ile\Pro-OH vs IRT-GSP	1.178258029262923	0.2386947037063701
605	Orn vs s-TSH	0.11436363588509317	0.9089496169936696
606	Orn vs IRT-GSP	1.1854436104079138	0.2358430511747007
670	MET vs IRT-GSP	0.8604502953376623	0.38954156967112075
732	PHE vs s-TSH	-0.08482307934239781	0.932402102546563
733	PHE vs IRT-GSP	1.1723815324679496	0.2410448559926427
794	TYR vs s-TSH	0.08317357727049521	0.9337135786295839
795	TYR vs IRT-GSP	1.1053697119429962	0.26900032252098255
801	TYR vs 2OH GA	1.9263572994232652	0.05406073149475504
803	TYR vs 3OH PA	1.9578791950245484	0.05024512758118209
811	HCYS vs Val	-1.7269901118878817	0.08417048240126618
834	HCYS vs C16	1.7509451078236844	0.07995640140462563
872	Pro vs BTD	-1.5373551801025696	0.12420716458437897
915	Pro vs s-TSH	0.16576699273718	0.8683404362415225
916	Pro vs IRT-GSP	1.1517073527016297	0.2494422819322853
974	Val vs s-TSH	0.1171323818903238	0.906755229034335

975	Val vs IRT-GSP	0.9308497676503816	0.3519320424138035
979	Val vs EMA	1.747592349778695	0.08053567769362635
980	Val vs GA	1.7415413410746086	0.08158977192463719
981	Val vs 2OH GA	1.5998487728329014	0.10963320298988123
982	Val vs 3OH GA	1.7546132744954857	0.0793265136906927
983	Val vs 3OH PA	1.6149095053476032	0.10633135913641768
985	Val vs OROTICO	1.7540274562141058	0.07942683722459258
986	Val vs PIVA	1.7581020638714457	0.07873117788464559
987	Val vs 2MBC	1.7572356519279655	0.07887868424324736
988	Val vs c4-b	1.756534012779795	0.07899830277279392
989	Val vs c4-i	1.7567203138562253	0.07896652696639712
1090	C0 vs IRT-GSP	0.8553742382089448	0.39234469450000575
1214	C4OH\C3DC vs c4-b	0.808440296105122	0.41883781225729577
1427	C5DC\C6OH vs C6DC	1.0628846336251743	0.2878347073164768
1977	C14:2 vs D-ADO	1.3452146704001855	0.17855688703160738
2404	C24 vs D-ADO	1.7803969501873016	0.07501846760647048
2675	3OH GA vs OROTICO	-1.563174530694754	0.11816959671630518
2700	c4-b vs c4-i	1.2488467095774547	0.21186715353917676

Table 2.6: risultati t-test tra variabili con p-value > 0.05

Per quanto riguarda le coppie di features riportate nella tabella precedente [2.6], la differenza osservata tra le medie dei due gruppi non è statisticamente significativa, dunque non è possibile rifiutare l'ipotesi nulla di uguaglianza tra le distribuzioni.

Un secondo test utilizzato per confrontare le distribuzioni di variabili continue è il test di Kolmogorov-Smirnov: si tratta di un test simile al t-test, che permette di stabilire il grado di somiglianza di due distribuzioni. Si basa sulla differenza massima assoluta osservata nelle funzioni di distribuzione cumulativa relativa ai campioni estratti da due popolazioni. In particolare, rispetto al t-test, può essere utile per individuare differenze all'interno di distribuzioni con medie simili ma varianze molto differenti.

	VARIABLE	STATISTIC	P-VALUE
154	Allele 1 vs HCYS	0.0033238992753696697	0.07603901070907249
200	Allele 1 vs s-TSH	0.0013491717302873837	0.950291290049963
201	Allele 1 vs IRT-GSP	0.0008994478201915891	0.999753430076964
203	Allele 1 vs s-17OHP	0.0013897483236794855	0.937217979122844
205	Allele 1 vs EMA	0.003357713103196421	0.07112350140235968
206	Allele 1 vs GA	0.0033644758687617713	0.07017349759014102
207	Allele 1 vs 2OH GA	0.002938421638144703	0.15524340727242436
208	Allele 1 vs 3OH GA	0.0033847641654578225	0.06738798128779044
209	Allele 1 vs 3OH PA	0.0031041093944957853	0.11547751510143334
210	Allele 1 vs MCA	0.001998397224561012	0.5954829671487314
211	Allele 1 vs OROTICO	0.0033746200171097967	0.06876873453947174
212	Allele 1 vs PIVA	0.0033881455482404977	0.06693301124893791
213	Allele 1 vs 2MBC	0.0033847641654578225	0.06738798128779044
214	Allele 1 vs c4-b	0.0033881455482404977	0.06693301124893791
215	Allele 1 vs c4-i	0.0033847641654578225	0.06738798128779044
855	HCYS vs s-TSH	0.0033644758687617717	0.07017349759014102
856	HCYS vs IRT-GSP	0.003310373744238969	0.07808422567163298
858	HCYS vs s-17OHP	0.0033881455482404977	0.06693301124893791
860	HCYS vs EMA	0.0026543854843999906	0.24802443089989235
861	HCYS vs GA	0.0017481748986430512	0.7561172142782816
862	HCYS vs 2OH GA	0.003012812059363556	0.13620619408086632
863	HCYS vs 3OH GA	0.0032528902369334915	0.08730242033136526

864	HCYS vs 3OH PA	0.0023027216750017756	0.4124723475502725
865	HCYS vs MCA	0.0027186317572708185	0.2240578507085068
866	HCYS vs OROTICO	0.0032799412991948926	0.08285644934759995
867	HCYS vs PIVA	0.003367857251544447	0.0697025559723029
868	HCYS vs 2MBC	0.003361094485979096	0.07064714214075751
869	HCYS vs c4-b	0.003279941299194893	0.08285644934759995
870	HCYS vs c4-i	0.0033441875720657205	0.07305632193260214
2324	C18:2OH vs C22	0.001399892472027514	0.9336708043483329
2581	s-TSH vs IRT-GSP	0.0011530515288922251	0.9892393707628019
2583	s-TSH vs s-17OHP	4.057659339210177e-05	1.0
2585	s-TSH vs EMA	0.003378001399892472	0.06830583128594236
2586	s-TSH vs GA	0.003378001399892472	0.06830583128594236
2587	s-TSH vs 2OH GA	0.003350950337631071	0.07208441128741361
2588	s-TSH vs 3OH GA	0.0033847641654578225	0.06738798128779044
2589	s-TSH vs 3OH PA	0.003350950337631071	0.07208441128741361
2590	s-TSH vs MCA	0.003344187572065721	0.07305632193260214
2591	s-TSH vs OROTICO	0.0033746200171097967	0.06876873453947174
2592	s-TSH vs PIVA	0.0033881455482404977	0.06693301124893791
2593	s-TSH vs 2MBC	0.0033847641654578225	0.06738798128779044
2594	s-TSH vs c4-b	0.0033881455482404977	0.06693301124893791
2595	s-TSH vs c4-i	0.0033847641654578225	0.06738798128779044
2597	IRT-GSP vs s-17OHP	0.0011902467395016518	0.9846801709342924
2599	IRT-GSP vs EMA	0.003354331720413746	0.07160258717640466
2600	IRT-GSP vs GA	0.003361094485979096	0.07064714214075751
2601	IRT-GSP vs 2OH GA	0.002208042957086871	0.46607113400019495
2602	IRT-GSP vs 3OH GA	0.0033847641654578225	0.06738798128779044
2603	IRT-GSP vs 3OH PA	0.002268907847175024	0.4311995069553345
2604	IRT-GSP vs MCA	0.002197898808738846	0.4720211243049146
2605	IRT-GSP vs OROTICO	0.0033712386343271216	0.06923430553246746
2606	IRT-GSP vs PIVA	0.0033847641654578225	0.06738798128779044
2607	IRT-GSP vs 2MBC	0.0033813827826751474	0.06784558408847163
2608	IRT-GSP vs c4-b	0.0033847641654578225	0.06738798128779044
2609	IRT-GSP vs c4-i	0.0033813827826751474	0.06784558408847163
2624	s-17OHP vs EMA	0.0033881455482404977	0.06693301124893791
2625	s-17OHP vs GA	0.0033881455482404977	0.06693301124893791
2626	s-17OHP vs 2OH GA	0.0033881455482404977	0.06693301124893791
2627	s-17OHP vs 3OH GA	0.0033881455482404977	0.06693301124893791
2628	s-17OHP vs 3OH PA	0.0033881455482404977	0.06693301124893791
2629	s-17OHP vs MCA	0.0033847641654578225	0.06738798128779044
2630	s-17OHP vs OROTICO	0.0033881455482404977	0.06693301124893791
2631	s-17OHP vs PIVA	0.0033881455482404977	0.06693301124893791
2632	s-17OHP vs 2MBC	0.0033881455482404977	0.06693301124893791
2633	s-17OHP vs c4-b	0.0033881455482404977	0.06693301124893791
2634	s-17OHP vs c4-i	0.0033881455482404977	0.06693301124893791
2646	EMA vs GA	0.0016534961807281473	0.8129496275962604
2647	EMA vs 2OH GA	0.003222457791889415	0.0925418239895569
2648	EMA vs 3OH GA	0.0020829317941278907	0.5418736851471606
2649	EMA vs 3OH PA	0.003144685987887887	0.1071363259201844
2650	EMA vs MCA	0.0027186317572708185	0.2240578507085068
2651	EMA vs OROTICO	0.0023703493306552782	0.37648692316736054
2652	EMA vs PIVA	0.0033238992753696697	0.07603901070907249
2653	EMA vs 2MBC	0.0032258391746720905	0.09194697418415865
2654	EMA vs c4-b	0.0028538870685778242	0.17939733420074877
2655	EMA vs c4-i	0.0031379232223225366	0.10849085760421462
2656	GA vs 2OH GA	0.0032326019402374407	0.09076687003087236
2657	GA vs 3OH GA	0.002904607810317951	0.16457349002138055
2658	GA vs 3OH PA	0.0030195748249289066	0.1345736784398135
2659	GA vs MCA	0.0027186317572708185	0.2240578507085068
2660	GA vs OROTICO	0.00299590514545018	0.1403577979175219

2661	GA vs PIVA	0.0033374248065003703	0.07403932838819105
2662	GA vs 2MBC	0.0032664157680641925	0.08505500904872332
2663	GA vs c4-b	0.0030432445044076325	0.12898457670879826
2664	GA vs c4-i	0.0032292205574547656	0.09135532704377392
2665	2OH GA vs 3OH GA	0.003357713103196421	0.07112350140235968
2666	2OH GA vs 3OH PA	0.0008081504850593602	0.9999763701186174
2667	2OH GA vs MCA	0.0027186317572708185	0.2240578507085068
2668	2OH GA vs OROTICO	0.003354331720413746	0.07160258717640466
2669	2OH GA vs PIVA	0.003367857251544447	0.0697025559723029
2670	2OH GA vs 2MBC	0.0033644758687617717	0.07017349759014102
2671	2OH GA vs c4-b	0.003367857251544447	0.0697025559723029
2672	2OH GA vs c4-i	0.0033644758687617717	0.07017349759014102
2673	3OH GA vs 3OH PA	0.0033374248065003703	0.07403932838819105
2674	3OH GA vs MCA	0.0027727338817936206	0.20527163904345236
2675	3OH GA vs OROTICO	0.0005241143313146478	0.9999999999990894
2676	3OH GA vs PIVA	0.0032427460885854666	0.08902051459373306
2677	3OH GA vs 2MBC	0.0022553823160443233	0.43882227254041994
2678	3OH GA vs c4-b	0.001670403094641523	0.8031291218121117
2679	3OH GA vs c4-i	0.001761700429773752	0.7476885926304546
2680	3OH PA vs MCA	0.0027186317572708185	0.2240578507085068
2681	3OH PA vs OROTICO	0.0033374248065003703	0.07403932838819105
2682	3OH PA vs PIVA	0.003374620017109797	0.06876873453947174
2683	3OH PA vs 2MBC	0.003350950337631071	0.07208441128741361
2684	3OH PA vs c4-b	0.003350950337631071	0.07208441128741361
2685	3OH PA vs c4-i	0.003347568954848396	0.0725689855827848
2686	MCA vs OROTICO	0.0027761152645762958	0.2041389741509646
2687	MCA vs PIVA	0.003306992361456294	0.07860273433639453
2688	MCA vs 2MBC	0.0031649742845839377	0.10315648896942742
2689	MCA vs c4-b	0.0027727338817936206	0.20527163904345236
2690	MCA vs c4-i	0.0030128120593635563	0.1362061940808661
2691	OROTICO vs PIVA	0.0032528902369334915	0.08730242033136526
2692	OROTICO vs 2MBC	0.002735538671184194	0.2180519432399588
2693	OROTICO vs c4-b	0.0021708477464774446	0.4880693863198473
2694	OROTICO vs c4-i	0.0022181871054348964	0.4601594672149335
2695	PIVA vs 2MBC	0.003144685987887887	0.1071363259201844
2696	PIVA vs c4-b	0.003131160456757186	0.10985951936133598
2697	PIVA vs c4-i	0.0031142535428438107	0.11334366468401813
2698	2MBC vs c4-b	0.0006898020876657301	0.999997576567045
2699	2MBC vs c4-i	0.0011158563182827987	0.9927239831106534
2700	c4-b vs c4-i	0.0004936818862705714	0.9999999999999791

Table 2.7: Kolmogorov-Smirnov test tra coppie di variabili con p-value > 0.05

Come per quanto riguarda i t-test, sono state riportate, nella tabella [2.7] tutte le coppie di variabili per cui i p-values ottenuti col test di Kolmogorov-Smirnov sono superiori alla soglia di significatività di 0.05, dunque con differenza osservata tra i due gruppi non statisticamente significativa: per tutte queste coppie di variabili non posso rifiutare l'ipotesi nulla di uguaglianza tra le distribuzioni.

2.3 Risultati e discussione analisi per dati stratificati

La seguente analisi ha l’obiettivo di esplorare ulteriormente la distribuzione della variabile Reparto, e la relazione che intercorre tra questa e le altre features del dataset, per verificare se ci sono evidenze di differenze significative tra i gruppi di osservazioni che facevano parte dei tre tipi di reparto indicati: “generico”, “neo-patologico” e “nido”.

2.3.1 Statistiche descrittive variabili quantitative

La prima tabella [2.8] indica le statistiche descrittive delle variabili quantitative per valori della variabile *Reparto* uguali a “Nido” (ovvero per cui il campione proviene da un reparto “Nido” di uno degli ospedali lombardi); si tratta di 225284 osservazioni sulle 295737 totali. Per medie e quantili non vengono riportati gli intervalli di confidenza al 95%, in quanto si tratta di valori sempre molto vicini alle statistiche stesse (nell’ordine di <0.001%), per cui l’intervallo di confidenza riportato sarebbe un’informazione overkilling:

	unità misura	count	miss %	min	2.5%	25%	50%	75%	97.5%	max	iqr	skew	kurt	mean	std	outlier %
ASATotal	µmol/L	17358	92%	0.03	0.11	0.18	0.24	0.32	0.72	2.77	0.14	4.05	28.59	0.28	0.18	5%
Ala	µmol/L	224850	0%	0.0	146.5	214.53	261.7	321.89	486.14	998529	107.36	88.35	10480.52	347.93	5053.0	5%
Allele 1		312	100%	4.0	9.0	31.0	31.0	56.25	194.0	224.0	25.25	3.0	10.32	46.38	36.98	4%
Arg	µmol/L	224850	0%	0.0	1.48	5.01	8.69	14.19	32.68	4946.0	9.18	129.0	22923.1	10.96	21.01	5%
Cit	µmol/L	224850	0%	0.0	6.1	11.06	14.13	17.9	29.12	6032.4	6.84	108.4	14528.52	15.45	35.37	5%
Glu	µmol/L	17358	92%	29.2	130.2	188.19	225.8	271.93	389.64	646.13	83.73	0.91	1.65	235.03	66.99	5%
Gly	µmol/L	224850	0%	0.0	174.0	309.01	390.1	483.09	747.18	926743.3	174.08	75.84	6379.8	523.4	7812.4	5%
Leu\Ile\Pro-OH	µmol/L	224849	0%	0.0	89.9	123.66	147.2	176.97	258.45	298307.4	53.31	76.98	6230.5	187.8	2449.8	5%
Orn	µmol/L	224850	0%	0.0	50.74	79.55	101.3	131.06	230.05	287239.5	51.51	144.1	24480.62	122.5	1278.1	5%
MET	µmol/L	224849	0%	0.0	10.44	16.07	19.68	23.82	34.4	6565.84	7.75	101.4	13489.08	20.77	34.13	5%
PHE	µmol/L	224850	0%	0.12	34.66	46.61	53.93	62.37	84.38	123132.0	15.76	389.4	171022.1	57.43	277.97	5%
TYR	µmol/L	224850	0%	0.08	47.74	72.84	91.12	115.18	192.04	352349.8	42.34	147.9	24879.58	113.35	1713.0	5%
HCYS	µM	775	100%	0.0	1.07	2.2	2.86	3.85	6.73	76.3	1.65	17.49	405.2	3.28	3.09	5%
Pro	µmol/L	224850	0%	0.0	112.4	152.02	178.2	210.48	299.36	484459.1	58.46	72.97	6059.04	237.08	3380.0	5%
Val	µmol/L	224850	0%	0.0	77.2	110.96	133.2	160.28	229.05	361190.7	49.32	88.65	9071.51	168.11	2255.4	5%
BTD	U/dl	81892	64%	11.0	131.9	213.41	260.8	300.78	353.88	486.7	87.37	-0.35	-0.39	255.13	59.64	5%
C0	µmol/L	224850	0%	0.0	8.27	13.92	18.49	24.33	40.02	4250.44	10.41	68.56	5544.62	20.45	34.97	5%
C3	µmol/L	224850	0%	0.0	0.47	1.21	1.77	2.4	4.27	77.52	1.19	9.37	484.37	1.9	1.1	5%
C4OHC3DC	µmol/L	224850	0%	0.0	0.05	0.11	0.17	0.24	0.4	3.63	0.13	1.33	12.37	0.18	0.09	4%
C4	µmol/L	224850	0%	0.0	0.09	0.16	0.21	0.28	0.56	10.38	0.12	6.78	365.18	0.24	0.13	5%
C5OHC4DC	µmol/L	224850	0%	0.0	0.1	0.16	0.2	0.25	0.37	11.0	0.09	32.02	3166.36	0.21	0.09	4%
C5	µmol/L	224850	0%	0.0	0.05	0.09	0.11	0.14	0.26	10.92	0.05	46.68	6486.05	0.12	0.07	3%
C5:1	µmol/L	224850	0%	0.0	0.0	0.01	0.01	0.01	0.02	0.4	0.0	6.51	171.58	0.01	0.01	1%
C5DC\C6OH	µmol/L	224850	0%	0.0	0.05	0.09	0.12	0.15	0.23	15.22	0.06	77.03	19656.84	0.12	0.06	5%
C6	µmol/L	224850	0%	0.0	0.02	0.03	0.04	0.05	0.09	2.09	0.02	8.27	571.04	0.05	0.02	3%
C6DC	µmol/L	224850	0%	0.0	0.04	0.09	0.12	0.15	0.24	15.53	0.06	69.72	17152.44	0.12	0.06	5%
C8	µmol/L	224850	0%	0.0	0.02	0.04	0.05	0.07	0.13	37.58	0.03	325.9	130657.3	0.06	0.09	3%
C8:1	µmol/L	224850	0%	0.0	0.01	0.02	0.03	0.05	0.16	1.39	0.03	3.65	27.05	0.04	0.04	3%
C10	µmol/L	224850	0%	0.0	0.03	0.06	0.08	0.11	0.21	2.51	0.05	3.7	60.58	0.09	0.05	3%
C10:1	µmol/L	224850	0%	0.0	0.02	0.04	0.05	0.06	0.09	0.76	0.02	2.2	29.4	0.05	0.02	3%
C10:2	µmol/L	224850	0%	0.0	0.0	0.0	0.0	0.0	0.01	0.81	0.0	27.62	3371.16	0.0	0.01	2%
C12	µmol/L	224850	0%	0.0	0.03	0.07	0.1	0.14	0.29	11.18	0.07	17.29	2336.57	0.11	0.07	3%
C12:1	µmol/L	224850	0%	0.0	0.02	0.04	0.07	0.11	0.23	1.5	0.07	2.1	10.56	0.08	0.06	4%
C14	µmol/L	224850	0%	0.0	0.06	0.16	0.21	0.27	0.44	4.01	0.11	1.32	18.83	0.22	0.1	4%
C14:1	µmol/L	224850	0%	0.0	0.02	0.07	0.1	0.15	0.3	5.07	0.08	3.51	114.81	0.12	0.08	4%
C14:2	µmol/L	224850	0%	0.0	0.01	0.01	0.02	0.02	0.04	0.62	0.01	5.71	219.18	0.02	0.01	3%
C14-OH	µmol/L	224850	0%	0.0	0.0	0.01	0.01	0.02	0.04	0.45	0.01	2.0	27.05	0.02	0.01	2%
C16	µmol/L	224850	0%	0.0	0.54	2.31	3.28	4.22	6.44	66.28	1.91	3.63	101.7	3.26	1.65	5%
C16:1	µmol/L	224850	0%	0.0	0.02	0.15	0.23	0.3	0.46	11.03	0.15	3.5	295.1	0.22	0.12	5%
C16-OH	µmol/L	224850	0%	0.0	0.01	0.01	0.02	0.03	0.05	2.16	0.02	35.66	5535.2	0.02	0.01	3%
C16:1-OH	µmol/L	224850	0%	0.0	0.01	0.03	0.04	0.05	0.08	0.86	0.02	1.41	33.91	0.04	0.02	2%
C18	µmol/L	224850	0%	0.0	0.29	0.72	0.97	1.24	1.92	32.85	0.52	8.51	379.3	1.0	0.47	5%

C18:1	µmol/L	224850	0%	0.0	0.59	1.19	1.55	1.95	2.93	43.54	0.76	10.49	469.4	1.61	0.68	5%
C18:2	µmol/L	224850	0%	0.0	0.06	0.11	0.16	0.22	0.42	1.98	0.11	2.07	9.06	0.18	0.1	5%
C18-OH	µmol/L	224850	0%	0.0	0.0	0.01	0.01	0.02	0.03	0.73	0.01	6.8	541.8	0.01	0.01	1%
C18:1-OH	µmol/L	224850	0%	0.0	0.01	0.02	0.02	0.03	0.05	1.38	0.01	14.87	1614.3	0.02	0.01	2%
C18:2OH	µmol/L	17358	92%	0.0	0.01	0.01	0.01	0.01	0.02	0.08	0.0	2.43	14.7	0.01	0.0	4%
C20	µmol/L	17358	92%	0.0	0.01	0.03	0.04	0.05	0.12	0.5	0.03	3.85	29.9	0.04	0.03	5%
C22	µmol/L	17358	92%	0.0	0.01	0.01	0.01	0.01	0.03	0.11	0.0	3.98	24.8	0.01	0.01	3%
C24	µmol/L	17358	92%	0.0	0.01	0.01	0.02	0.02	0.04	0.15	0.01	3.28	19.2	0.02	0.01	4%
C26	µmol/L	17358	92%	0.0	0.0	0.01	0.01	0.01	0.03	0.13	0.01	4.59	34.3	0.01	0.01	3%
SA	µmol/L	224851	0%	0.0	0.22	0.58	0.8	1.04	1.52	21.61	0.46	8.09	297.7	0.82	0.39	5%
ADO	µmol/L	17358	92%	0.08	0.26	0.43	0.56	0.71	1.11	2.74	0.27	1.36	4.6	0.59	0.22	5%
D-ADO	µmol/L	17358	92%	0.0	0.01	0.01	0.02	0.02	0.06	0.36	0.01	7.43	77.5	0.02	0.02	4%
C20:0-LPC	µmol/L	17358	92%	0.02	0.11	0.18	0.28	0.46	1.19	5.2	0.27	2.87	15.5	0.37	0.3	5%
C22:0-LPC	µmol/L	17358	92%	0.02	0.1	0.16	0.21	0.29	0.72	2.9	0.13	3.26	17.8	0.25	0.16	5%
C24:0-LPC	µmol/L	17358	92%	0.04	0.23	0.36	0.46	0.59	1.26	2.73	0.23	2.48	9.3	0.52	0.26	5%
C26:0-LPC	µmol/L	17358	92%	0.03	0.13	0.2	0.26	0.33	0.78	2.62	0.13	3.76	21.9	0.3	0.18	5%
s-TSH	µU/mL	8	100%	2.96	3.9	15.84	27.7	59.56	535.0	619.0	43.72	2.62	6.9	109.5	210.3	25%
IRT-GSP	ng/mL	318	100%	0.18	8.02	12.32	16.57	22.33	41.28	73.2	10.01	1.76	5.2	18.9	9.5	5%
TGAL	mg/dl	81829	64%	0.0	0.0	0.95	1.71	2.81	6.4	470.46	1.85	97.8	19342.5	2.11	2.35	3%
s-17OHP		0	100%													
MMA	µM	209640	7%	0.0	0.0	0.0	0.0	0.0	0.0	37.19	0.0	257.9	87413.0	0.0	0.1	0%
EMA	µM	775	100%	0.0	0.31	0.65	0.9	1.21	4.27	15.69	0.56	5.9	53.9	1.13	1.11	5%
GA	µM	775	100%	0.0	0.51	1.19	1.57	2.09	3.77	10.74	0.9	3.37	24.1	1.71	0.95	5%
2OH GA	µM	775	100%	0.0	5.33	10.11	15.14	22.2	35.61	51.67	12.09	0.91	0.9	16.84	8.69	5%
3OH GA	µM	775	100%	0.0	0.05	0.15	0.31	0.53	0.97	3.67	0.37	3.59	28.9	0.37	0.3	5%
3OH PA	µM	775	100%	0.2	2.39	5.35	15.19	22.07	30.33	55.5	16.72	0.31	-0.4	14.48	8.81	5%
MCA	µM	775	100%	0.0	0.21	4773.8	9052	13517.3	26047.1	115527.8	8743.5	3.39	39.23	9552.0	7992.7	5%
OROTICO	µM	775	100%	0.0	0.07	0.23	0.32	0.44	0.86	24.65	0.21	14.56	225.12	0.44	1.33	5%
PIVA	µM	775	100%	0.0	0.0	0.0	0.0	0.0	0.04	0.45	0.0	8.33	80.0	0.01	0.03	3%
2MBC	µM	775	100%	0.0	0.03	0.05	0.07	0.09	0.17	0.85	0.04	6.61	79.1	0.08	0.05	5%
c4-b	µM	775	100%	0.0	0.02	0.06	0.09	0.16	0.8	1.51	0.1	3.08	10.94	0.17	0.21	5%
c4-i	µM	775	100%	0.0	0.02	0.08	0.11	0.16	0.38	13.3	0.08	25.93	703.38	0.15	0.48	5%
Gestational Age	settim	225284	0%	23.0	35.0	38.0	39.0	40.0	41.0	43.0	2.0	-0.86	1.57	38.9	1.51	1%
Weight	g	225284	0%	620	2250	2950.0	3250	3550.0	4140.0	5000.0	600.0	-0.17	0.3	3241.1	471.7	5%

Table 2.8: tabella informazioni e statistiche descrittive variabili quantitative per Reparto = 'Nido'

La seconda tabella [2.9] indica invece le statistiche descrittive delle variabili quantitative con variabile Reparto pari a "Neo-patologico" (ovvero per cui il campione proviene da un reparto "Neo-patologico" di uno degli ospedali lombardi), per un totale di 29968 osservazioni sulle 295737 totali.

	unità misura	count	miss %	min	2.5%	25%	50%	75%	97.5%	max	iqr	skew	kurt	mean	std	outlier %
ASATotal	µmol/L	2424	92%	0.04	0.09	0.16	0.22	0.3	0.68	1.88	0.14	3.95	25.31	0.26	0.16	5%
Ala	µmol/L	29871	0%	0.0	124.3	206.27	264.64	336.34	561.17	793862.7	130.07	84.59	8154.92	379.22	6741.6	5%
Allele 1		50	100%	9.0	15.58	31.0	31.0	57.0	218.1	220.0	26.0	2.66	6.38	52.58	50.23	8%
Arg	µmol/L	29871	0%	0.0	1.75	6.84	13.12	23.82	62.82	5352.47	16.98	43.01	2172.77	20.13	78.1	5%
Cit	µmol/L	29871	0%	0.0	6.02	11.56	15.56	20.61	34.77	3118.48	9.05	65.96	5022.31	17.33	33.11	5%
Glu	µmol/L	2424	92%	50.5	95.59	151.55	193.69	250.12	383.63	694.9	98.57	1.01	1.7	206.42	76.2	5%
Gly	µmol/L	29871	0%	0.0	141.5	252.4	331.05	433.86	738.62	656552.8	181.46	53.48	2947.47	554.53	9989.9	5%
Leu\Ile\Pro-OH	µmol/L	29871	0%	0.43	79.95	124.96	157.54	200.49	320.9	348341.4	75.53	55.8	3227.56	260.74	4766.2	5%
Orn	µmol/L	29871	0%	0.0	43.16	80.4	107.92	144.85	269.73	401343.3	64.45	76.19	6432.86	178.16	3834.0	5%
MET	µmol/L	29871	0%	0.0	11.0	18.06	23.07	29.8	55.39	9288.54	11.74	56.67	3639.31	27.92	107.27	5%
PHE	µmol/L	29872	0%	7.64	28.81	41.86	51.51	64.77	111.36	197180.6	22.91	82.64	7203.31	82.2	1862.1	5%
TYR	µmol/L	29872	0%	0.0	30.89	63.64	85.71	115.62	235.28	112111.4	51.98	118.2	14448.4	107.3	862.58	5%
HCYS	µM	213	99%	0.78	0.87	1.67	2.47	3.4	7.36	10.69	1.73	1.89	5.0	2.8	1.64	6%
Pro	µmol/L	29871	0%	26.4	87.5	133.04	162.83	199.98	322.15	527548.2	66.95	62.04	4505.38	283.42	5561.9	5%
Val	µmol/L	29871	0%	21.5	59.62	98.84	127.44	165.01	283.0	760742.1	66.17	76.72	6752.34	258.62	7243.3	5%
BTD	U/dl	11131	63%	32.4	119.6	214.02	276.18	315.08	364.93	463.31	101.07	-0.51	-0.41	262.8	68.12	5%
C0	µmol/L	29871	0%	0.12	9.73	17.5	23.02	30.55	66.65	7978.04	13.05	69.0	5976.98	27.69	73.79	5%
C3	µmol/L	29871	0%	0.0	0.31	0.66	1.19	2.19	4.95	54.66	1.53	6.66	162.92	1.62	1.45	5%

C4OH\C3DC	μmol/L	29871	0%	0.0	0.03	0.06	0.09	0.13	0.26	1.95	0.07	3.28	40.5	0.1	0.06	3%
C4	μmol/L	29871	0%	0.0	0.08	0.14	0.19	0.27	0.67	11.98	0.13	18.49	896.64	0.24	0.2	5%
C5OH\C4DC	μmol/L	29871	0%	0.0	0.09	0.14	0.18	0.22	0.36	1.4	0.08	1.88	11.15	0.19	0.07	5%
C5	μmol/L	29871	0%	0.02	0.07	0.12	0.15	0.22	0.51	5.08	0.1	5.15	98.0	0.19	0.12	5%
C5:1	μmol/L	29871	0%	0.0	0.0	0.01	0.01	0.01	0.02	0.12	0.0	2.06	17.51	0.01	0.01	2%
C5DC\ C6OH	μmol/L	29871	0%	0.0	0.04	0.07	0.09	0.12	0.21	30.29	0.05	131.1	19955.9	0.1	0.19	4%
C6	μmol/L	29871	0%	0.0	0.02	0.03	0.04	0.05	0.1	1.65	0.02	14.11	667.85	0.04	0.03	5%
C6DC	μmol/L	29871	0%	0.0	0.03	0.06	0.09	0.12	0.2	2.69	0.06	7.92	282.2	0.09	0.05	5%
C8	μmol/L	29871	0%	0.0	0.02	0.04	0.05	0.06	0.15	10.96	0.02	91.58	11813.8	0.06	0.08	4%
C8:1	μmol/L	29871	0%	0.0	0.02	0.04	0.06	0.11	0.25	2.65	0.07	7.25	165.11	0.08	0.07	4%
C10	μmol/L	29871	0%	0.0	0.02	0.04	0.05	0.07	0.14	1.55	0.03	5.61	133.21	0.06	0.04	3%
C10:1	μmol/L	29871	0%	0.0	0.02	0.03	0.04	0.06	0.11	1.12	0.03	7.72	148.25	0.05	0.03	4%
C10:2	μmol/L	29871	0%	0.0	0.0	0.0	0.0	0.01	0.02	7.66	0.01	163.9	27838.2	0.01	0.05	2%
C12	μmol/L	29871	0%	0.0	0.02	0.04	0.05	0.07	0.14	2.87	0.03	16.36	1018.51	0.06	0.04	4%
C12:1	μmol/L	29871	0%	0.0	0.01	0.02	0.03	0.04	0.1	1.24	0.02	6.96	162.55	0.04	0.03	3%
C14	μmol/L	29871	0%	0.01	0.04	0.09	0.13	0.19	0.35	7.24	0.1	20.16	1130.16	0.15	0.1	4%
C14:1	μmol/L	29871	0%	0.0	0.01	0.03	0.05	0.07	0.16	11.42	0.04	87.93	11087.9	0.06	0.08	3%
C14:2	μmol/L	29871	0%	0.0	0.0	0.01	0.02	0.02	0.04	0.71	0.01	15.66	701.27	0.02	0.01	2%
C14-OH	μmol/L	29871	0%	0.0	0.0	0.01	0.01	0.01	0.02	0.32	0.0	10.91	443.9	0.01	0.01	2%
C16	μmol/L	29871	0%	0.06	0.35	0.76	1.31	2.43	4.92	43.95	1.67	4.2	85.98	1.74	1.35	5%
C16:1	μmol/L	29871	0%	0.0	0.02	0.04	0.08	0.16	0.35	4.38	0.12	5.11	152.81	0.11	0.1	3%
C16-OH	μmol/L	29871	0%	0.0	0.0	0.01	0.01	0.02	0.03	2.34	0.01	85.89	8598.58	0.01	0.02	2%
C16:1-OH	μmol/L	29871	0%	0.0	0.01	0.02	0.03	0.04	0.07	0.61	0.02	4.26	96.38	0.03	0.02	2%
C18	μmol/L	29871	0%	0.0	0.18	0.4	0.61	0.89	1.57	21.82	0.49	9.12	290.95	0.68	0.44	5%
C18:1	μmol/L	29871	0%	0.04	0.39	0.79	1.14	1.59	2.71	32.72	0.8	7.45	223.79	1.25	0.7	5%
C18:2	μmol/L	29871	0%	0.01	0.08	0.16	0.24	0.36	0.88	11.14	0.2	10.98	404.33	0.3	0.24	4%
C18-OH	μmol/L	29871	0%	0.0	0.0	0.0	0.01	0.01	0.02	3.38	0.01	123.1	17899.2	0.01	0.02	1%
C18:1-OH	μmol/L	29871	0%	0.0	0.01	0.01	0.02	0.02	0.04	1.59	0.01	51.63	3943.42	0.02	0.02	4%
C18:2OH	μmol/L	2424	92%	0.0	0.01	0.01	0.01	0.02	0.03	0.05	0.01	1.78	4.59	0.01	0.01	5%
C20	μmol/L	2424	92%	0.01	0.01	0.02	0.02	0.03	0.08	0.28	0.02	3.59	24.34	0.03	0.02	5%
C22	μmol/L	2424	92%	0.0	0.0	0.01	0.01	0.01	0.03	0.08	0.0	3.72	23.55	0.01	0.01	3%
C24	μmol/L	2424	92%	0.0	0.0	0.01	0.01	0.02	0.04	0.11	0.01	2.54	12.76	0.02	0.01	4%
C26	μmol/L	2424	92%	0.0	0.0	0.01	0.01	0.01	0.03	0.11	0.01	4.11	29.7	0.01	0.01	4%
SA	μmol/L	29872	0%	0.0	0.21	0.57	0.79	1.02	1.5	11.59	0.45	7.67	189.55	0.81	0.4	5%
ADO	μmol/L	2424	92%	0.12	0.29	0.54	0.72	1.03	2.6	8.39	0.5	3.75	23.48	0.9	0.65	5%
D-ADO	μmol/L	2424	92%	0.0	0.01	0.01	0.02	0.02	0.04	0.25	0.01	7.54	81.63	0.02	0.01	5%
C20:0-LPC	μmol/L	2424	92%	0.06	0.13	0.27	0.41	0.63	1.49	3.94	0.36	2.72	12.68	0.51	0.38	5%
C22:0-LPC	μmol/L	2424	92%	0.05	0.1	0.17	0.23	0.33	0.81	3.6	0.16	4.4	42.06	0.29	0.2	5%
C24:0-LPC	μmol/L	2424	92%	0.1	0.19	0.32	0.42	0.55	1.17	2.29	0.24	2.3	8.4	0.48	0.25	5%
C26:0-LPC	μmol/L	2424	92%	0.05	0.1	0.19	0.25	0.35	0.79	1.99	0.17	3.0	14.88	0.3	0.19	5%
s-TSH	μU/mL	4	100%	8.59	8.82	10.92	17.9	26.79	34.03	34.84	15.87	0.58	-2.21	19.8	12.06	50%
IRT-GSP	ng/mL	24	100%	5.09	5.9	13.43	15.96	19.08	78.36	115.4	5.65	3.73	15.45	21.54	22.02	8%
TGAL	mg/dl	11125	63%	0.0	0.11	0.99	1.73	2.86	6.15	16.74	1.87	1.57	4.2	2.13	1.6	5%
s-17OHP		1	100%	487	487	487	487	487	487	487	0		487		0%	
MMA	μM	27812	7%	0.0	0.0	0.0	0.0	0.0	0.0	12.88	0.0	77.97	7490.76	0.0	0.12	1%
EMA	μM	213	99%	0.0	0.24	0.58	0.79	1.05	1.74	8.34	0.47	6.69	61.87	0.9	0.72	6%
GA	μM	213	99%	0.0	0.33	1.05	1.47	2.05	3.75	4.63	1.0	0.9	0.74	1.63	0.86	6%
2OH GA	μM	213	99%	2.01	2.85	7.46	11.09	17.62	35.7	62.87	10.16	1.9	6.27	13.28	8.74	6%
3OH GA	μM	213	99%	0.03	0.06	0.16	0.28	0.48	0.83	0.96	0.32	0.8	-0.18	0.33	0.22	6%
3OH PA	μM	213	99%	0.3	2.36	4.99	15.75	22.59	30.39	57.28	17.6	0.41	0.47	14.87	9.3	6%
MCA	μM	213	99%	0.0	0.23	4858.7	9729.8	15766.7	27894	35701.6	10908	0.54	-0.05	10457.7	7861.6	6%
OROTICO	μM	213	99%	0.0	0.08	0.2	0.32	0.47	0.86	1.08	0.27	1.04	1.21	0.36	0.2	6%
PIVA	μM	213	99%	0.0	0.0	0.0	0.0	0.01	0.06	0.99	0.01	11.59	145.84	0.02	0.07	3%
2MBC	μM	213	99%	0.0	0.03	0.06	0.1	0.17	0.54	5.22	0.1	12.2	165.61	0.16	0.37	6%
c4-b	μM	213	99%	0.02	0.03	0.07	0.1	0.17	0.81	1.32	0.1	2.77	8.3	0.18	0.22	6%
c4-i	μM	213	99%	0.0	0.03	0.06	0.11	0.17	0.6	0.87	0.11	2.83	9.75	0.15	0.14	6%
Gestational Age	settim	29968	0%	23.0	26.0	33.0	35.0	38.0	41.0	42.0	5.0	-0.58	-0.22	35.18	3.98	2%
Weight	g	29968	0%	350	760.0	1715.0	2340.0	3100.0	4000.0	5000.0	1385.0	0.07	-0.76	2384.72	889.42	5%

Table 2.9: tabella informazioni e statistiche descrittive variabili quantitative per Reparto = 'Neo-Patologico'

Infine, la terza e ultima tabella [2.10] indica le statistiche descrittive delle variabili quantitative con variabile *Reparto* indicata come “*Generico*” (ovvero per cui il campione non proviene da un reparto “Nido”,

né da un reparto “Neo-Patologico” di uno degli ospedali lombardi), per un totale di 40485 osservazioni sulle 295737 totali.

	unità misura	count	miss%	min	2.5%	25%	50%	75%	97.5%	max	iqr	skew	kurt	mean	std	outlier %
ASATotal	µmol/L	598	99%	0.05	0.1	0.18	0.23	0.32	0.75	2.36	0.14	4.56	34.05	0.27	0.19	5%
Ala	µmol/L	40336	0%	0.0	138	199.17	241.66	295.85	449.37	483425.3	96.68	85.8	7877.01	308.0	4190.2	5%
Allele 1		49	100%	2.0	5.6	29.0	31.0	46.0	74.0	75.0	17.0	0.53	0.46	36.86	16.12	6%
Arg	µmol/L	40336	0%	0.0	1.75	5.27	8.79	14.17	32.43	3222.58	8.9	125.1	20364.8	11.04	19.06	5%
Cit	µmol/L	40336	0%	0.0	5.84	10.03	12.71	16.11	26.88	1816.25	6.08	83.66	8108.4	13.8	16.44	5%
Glu	µmol/L	598	99%	49.4	128	179.9	216.99	268.83	389.87	532.2	88.94	0.87	1.14	228.05	67.19	5%
Gly	µmol/L	40336	0%	0.0	187	329.3	413.96	513.63	816.34	580454.9	184.31	108.3	12131.8	478.72	4548.4	5%
Leu\Ile\Pro-OH	µmol/L	40336	0%	1.27	84.4	115.94	138.2	166.36	245.33	317359.9	50.42	91.38	9329.98	176.03	2488.8	5%
Orn	µmol/L	40336	0%	0.59	53.7	83.82	106.26	136.97	243.6	176138.2	53.15	95.67	9424.47	134.11	1561.9	5%
MET	µmol/L	40336	0%	0.21	9.42	14.21	17.55	21.53	31.96	2622.7	7.32	92.04	9041.81	18.59	24.52	5%
PHE	µmol/L	40338	0%	14.2	32.3	42.75	49.36	57.16	77.99	8676.83	14.41	127.2	18396.4	51.35	53.97	5%
TYR	µmol/L	40338	0%	18.9	41.8	63.6	79.88	101.18	169.65	120122.0	37.58	188.2	36755.5	90.67	611.82	5%
HCYS	µM	14	100%	1.63	1.68	2.08	2.9	3.23	4.85	4.99	1.15	0.64	-0.17	2.94	1.02	14%
Pro	µmol/L	40336	0%	0.0	114.3	152.9	177.86	208.6	295.84	400411.1	55.63	113.94	15013.2	213.68	2631.4	5%
Val	µmol/L	40336	0%	1.13	74.1	105.64	126.47	151.71	217.46	245252.5	46.07	64.87	4359.36	178.52	2914.8	5%
BTD	U/dl	3035	93%	14.5	133	224.12	271.68	310.42	357.03	423.48	86.3	-0.55	-0.05	264.09	59.97	5%
C0	µmol/L	40336	0%	0.34	7.41	12.41	16.48	21.82	36.9	2920.76	9.41	97.83	12562.8	18.13	19.94	5%
C3	µmol/L	40336	0%	0.0	0.43	1.07	1.51	2.04	3.65	32.79	0.97	3.87	91.06	1.64	0.87	5%
C4OHIC3DC	µmol/L	40336	0%	0.0	0.04	0.09	0.14	0.2	0.35	0.87	0.11	1.18	2.45	0.15	0.08	4%
C4	µmol/L	40336	0%	0.0	0.07	0.14	0.19	0.26	0.51	2.36	0.12	2.54	15.79	0.22	0.12	5%
C5OHIC4DC	µmol/L	40336	0%	0.0	0.08	0.13	0.16	0.2	0.3	26.76	0.07	150.3	27315.8	0.17	0.15	4%
C5	µmol/L	40336	0%	0.0	0.05	0.07	0.1	0.13	0.26	2.97	0.06	6.29	152.07	0.11	0.06	5%
C5:1	µmol/L	40336	0%	0.0	0.0	0.0	0.01	0.01	0.02	0.2	0.0	3.3	68.84	0.01	0.01	1%
C5DC\C6OH	µmol/L	40336	0%	0.0	0.05	0.09	0.11	0.14	0.22	0.77	0.05	1.21	4.97	0.12	0.04	4%
C6	µmol/L	40336	0%	0.0	0.02	0.03	0.04	0.05	0.09	3.28	0.02	53.83	6826.7	0.04	0.03	3%
C6DC	µmol/L	40336	0%	0.0	0.04	0.08	0.11	0.13	0.22	2.19	0.05	3.88	123.75	0.11	0.05	3%
C8	µmol/L	40336	0%	0.0	0.02	0.04	0.06	0.07	0.14	33.89	0.03	188.7	37097.2	0.06	0.17	5%
C8:1	µmol/L	40336	0%	0.0	0.01	0.02	0.03	0.05	0.15	0.46	0.03	3.12	14.75	0.04	0.04	4%
C10	µmol/L	40336	0%	0.0	0.03	0.06	0.08	0.1	0.2	2.29	0.04	4.98	133.2	0.09	0.05	3%
C10:1	µmol/L	40336	0%	0.0	0.02	0.04	0.05	0.06	0.1	0.79	0.02	5.88	121.51	0.05	0.02	2%
C10:2	µmol/L	40336	0%	0.0	0.0	0.0	0.01	0.01	0.02	0.08	0.01	0.78	2.89	0.01	0.01	0%
C12	µmol/L	40336	0%	0.0	0.03	0.06	0.09	0.13	0.26	0.92	0.07	1.88	6.42	0.1	0.06	3%
C12:1	µmol/L	40336	0%	0.0	0.02	0.04	0.06	0.1	0.21	1.01	0.06	2.75	19.39	0.08	0.05	4%
C14	µmol/L	40336	0%	0.01	0.06	0.15	0.2	0.25	0.4	0.8	0.1	0.79	1.66	0.2	0.09	5%
C14:1	µmol/L	40336	0%	0.0	0.03	0.07	0.1	0.14	0.27	0.91	0.07	1.72	6.03	0.11	0.06	5%
C14:2	µmol/L	40336	0%	0.0	0.01	0.01	0.02	0.02	0.04	0.27	0.01	2.08	26.53	0.02	0.01	2%
C14-OH	µmol/L	40336	0%	0.0	0.0	0.01	0.01	0.02	0.03	0.08	0.01	1.46	3.96	0.01	0.01	3%
C16	µmol/L	40336	0%	0.02	0.49	2.24	3.02	3.86	5.93	75.53	1.62	3.89	170.53	3.05	1.43	5%
C16:1	µmol/L	40336	0%	0.0	0.02	0.15	0.21	0.27	0.42	0.8	0.12	0.29	0.39	0.21	0.1	3%
C16-OH	µmol/L	40336	0%	0.0	0.01	0.01	0.02	0.02	0.04	0.1	0.01	1.06	2.57	0.02	0.01	4%
C16:1-OH	µmol/L	40336	0%	0.0	0.01	0.03	0.04	0.05	0.07	0.15	0.02	0.7	1.49	0.04	0.02	2%
C18	µmol/L	40336	0%	0.01	0.26	0.67	0.89	1.14	1.77	11.03	0.47	1.95	33.96	0.92	0.39	5%
C18:1	µmol/L	40336	0%	0.01	0.54	1.16	1.5	1.88	2.85	22.04	0.72	4.18	118.16	1.55	0.62	5%
C18:2	µmol/L	40336	0%	0.01	0.06	0.11	0.15	0.21	0.42	1.5	0.1	2.45	13.22	0.17	0.1	4%
C18-OH	µmol/L	40336	0%	0.0	0.0	0.01	0.01	0.02	0.03	0.15	0.01	0.95	7.87	0.01	0.01	0%
C18:1-OH	µmol/L	40336	0%	0.0	0.01	0.02	0.02	0.03	0.04	0.17	0.01	0.99	4.8	0.02	0.01	3%
C18:2OH	µmol/L	598	99%	0.0	0.01	0.01	0.01	0.01	0.02	0.07	0.0	4.28	42.15	0.01	0.0	5%
C20	µmol/L	598	99%	0.01	0.01	0.03	0.04	0.06	0.14	0.25	0.03	1.98	5.64	0.05	0.03	5%
C22	µmol/L	598	99%	0.0	0.01	0.01	0.01	0.01	0.03	0.1	0.0	5.61	57.09	0.01	0.01	4%
C24	µmol/L	598	99%	0.0	0.01	0.01	0.02	0.02	0.04	0.16	0.01	5.04	51.28	0.02	0.01	5%
C26	µmol/L	598	99%	0.0	0.0	0.01	0.01	0.01	0.03	0.16	0.01	8.32	101.81	0.01	0.01	3%
SA	µmol/L	40337	0%	0.0	0.16	0.55	0.7	0.91	1.4	16.6	0.36	6.94	249.8	0.73	0.34	5%
ADO	µmol/L	598	99%	0.21	0.29	0.46	0.56	0.71	1.04	1.39	0.26	0.86	1.0	0.6	0.2	5%
D-ADO	µmol/L	598	99%	0.0	0.01	0.01	0.02	0.02	0.07	0.34	0.01	8.3	101.91	0.02	0.02	4%
C20:0-LPC	µmol/L	598	99%	0.07	0.1	0.17	0.27	0.48	1.3	3.89	0.3	4.17	29.12	0.39	0.37	5%
C22:0-LPC	µmol/L	598	99%	0.05	0.09	0.16	0.21	0.28	0.76	1.74	0.13	3.08	15.02	0.25	0.17	5%
C24:0-LPC	µmol/L	598	99%	0.09	0.2	0.33	0.42	0.55	1.17	3.12	0.21	3.46	21.64	0.49	0.27	5%
C26:0-LPC	µmol/L	598	99%	0.05	0.11	0.18	0.23	0.29	0.82	2.51	0.11	5.15	40.24	0.27	0.19	5%
s-TSH	µU/mL	1	100%	8.95	8.95	8.95	8.95	8.95	8.95	8.95	0.0		8.95		0%	
IRT-GSP	ng/mL	10	100%	8.45	9.44	15.77	18.51	22.29	26.24	27.03	6.52	-0.21	-0.1	18.33	5.48	20%

TGAL	mg/dl	3028	93%	0.0	0.0	1.02	2.01	3.31	6.88	36.41	2.29	2.93	31.5	2.42	1.97	3%
s-17OHP		0	100%													
MMA	µM	39810	2%	0.0	0.0	0.0	0.0	0.0	0.0	0.62	0.0	77.17	6734.14	0.0	0.01	0%
EMA	µM	14	100%	0.34	0.35	0.51	0.95	1.32	2.25	2.61	0.81	1.31	2.34	0.99	0.62	14%
GA	µM	14	100%	0.84	0.86	1.08	1.54	1.93	2.89	3.03	0.85	0.84	-0.29	1.63	0.69	14%
2OH GA	µM	14	100%	7.38	7.95	10.43	15.38	22.86	32.65	35.33	12.43	0.85	0.15	17.22	8.14	14%
3OH GA	µM	14	100%	0.05	0.07	0.19	0.37	0.61	1.12	1.15	0.42	0.84	-0.14	0.46	0.34	14%
3OH PA	µM	14	100%	2.72	2.85	4.99	17.35	22.03	29.62	32.55	17.04	0.11	-1.3	14.7	9.82	14%
MCA	µM	14	100%	0.16	0.19	40666.9	9146.8	13634.	20508	21104.52	9567.6	0.15	-0.91	9282.8	6984.8	14%
																5
OROTICO	µM	14	100%	0.05	0.08	0.24	0.3	0.43	0.66	0.67	0.19	0.55	0.11	0.33	0.17	14%
PIVA	µM	14	100%	0.0	0.0	0.0	0.0	0.0	0.03	0.03	0.0	2.29	4.08	0.0	0.01	7%
2MBC	µM	14	100%	0.04	0.04	0.05	0.06	0.07	0.19	0.24	0.02	3.13	10.82	0.07	0.05	14%
c4-b	µM	14	100%	0.03	0.03	0.06	0.07	0.16	0.2	0.21	0.11	0.62	-1.4	0.1	0.06	14%
c4-i	µM	14	100%	0.02	0.03	0.09	0.13	0.19	0.26	0.28	0.1	0.22	-0.61	0.14	0.07	14%
GestationalAge	settim	40485	0%	23.0	35.0	38.0	39.0	40.0	41.0	43.0	2.0	-1.73	7.04	38.79	1.7	2%
Weight	g	40485	0%	370	2110	2950.0	3250.0	3550.0	4150.0	5000.0	600.0	-0.55	1.55	3229.5	503.4	5%

Table 2.10: tabella informazioni e statistiche descrittive variabili quantitative per Reparto = 'Generico'

Si tratta tuttavia di una visione d’insieme abbastanza ampia, con moltissime informazioni raggruppate e in cui è difficile ottenere insights significativi e rilevanti, quindi, come nel caso dell’intero dataset, per valutare meglio le distribuzioni di tutte le variabili quantitative, è stato scelto di procedere con la visualizzazione tramite istogrammi.

2.3.2 Grafici variabili quantitative

Come nel caso dell’intero dataset [2.2.2], sono stati costruiti gli istogrammi relativi alle variabili quantitative, stratificati per la variabile *Reparto* (in rosso le distribuzioni per *Reparto Nido*, in blu le distribuzioni per *Reparto Neo-patologico*, in verde le distribuzioni per *Reparto Generico*), con l’esclusione degli outliers con le stesse accortezze adottate in precedenza.

Anche in questo caso, non essendo comunque il focus principale di questo progetto di tesi, i grafici sono stati riportati nell’Appendice 2 [Figure analisi esplorative stratificate per reparto]. È bene ricordare, a proposito di questi grafici, che data la differente numerosità dei tre gruppi considerati non bisogna lasciarsi ingannare dalle differenze nelle frequenze delle variabili rispetto al reparto, mentre bisogna considerare il trend generale delle distribuzioni.

Molte variabili seguono l’andamento generale del dataset, con distribuzioni prevalentemente asimmetriche negative, ovvero che presentano gran parte delle osservazioni in corrispondenza di valori molto bassi: in particolare, la stratificazione permette di individuare distribuzioni generalmente molto più asimmetriche per i neonati dei reparti “Nido”, mentre distribuzioni più “schiacciate” per i reparti “Neo-Patologico” e “Generico”, con code maggiormente allungate verso destra.

Come nel caso generale, ovviamente, è facile notare la presenza di alcune variabili con un’altissima percentuale di valori mancanti; infine, come nel caso completo, le variabili *GestationalAge* e *Weight* presentano distribuzioni differenti rispetto alle altre del dataset, più asimmetriche positive ma comunque con code e massimi meno accentuati rispetto alle altre variabili.

2.3.3 Analisi variabili qualitative

Per quanto riguarda le variabili qualitative, sono state create le seguenti cross tabulazioni per verificare le distribuzioni delle variabili in base al *Reparto* (per ogni variabile qualitativa, viene riportata la variabile

Reparto sulle colonne e le categorie della variabile qualitativa di interesse sulle righe; tra parentesi vengono indicate le frequenze relative):

		Grouped by Reparto				
		Missing	Overall	Generico	Neo-Pat	Nido
Twins	0.0	155920	129257 (92.4)	32438 (97.5)	6572 (62.8)	90247 (93.9)
	1.0		10560 (7.6)	837 (2.5)	3888 (37.2)	5835 (6.1)
TooYoung	0.0	0	292480 (98.9)	40005 (98.8)	28781 (96.0)	223694 (99.3)
	1.0		3257 (1.1)	480 (1.2)	1187 (4.0)	1590 (0.7)
TPNCARNFeed	0	0	295364 (99.9)	40485 (100.0)	29599 (98.8)	225280 (100.0)
	1		373 (0.1)	///	369 (1.2)	4 (0.0)
SampleQuality	OK	295733	4 (100.0)	1 (100.0)	///	3 (100.0)
BabyFed	0.0	172326	759 (0.6)	17 (0.1)	504 (4.5)	238 (0.3)
	1.0		122652 (99.4)	28395 (99.9)	10664 (95.5)	83593 (99.7)
AntibioticsMother	0.0	201061	78482 (82.9)	2553 (85.8)	8482 (76.8)	67447 (83.6)
	1.0		16194 (17.1)	423 (14.2)	2556 (23.2)	13215 (16.4)
TPNFeed	0	0	292390 (98.9)	40385 (99.8)	26812 (89.5)	225193 (100.0)
	1		3347 (1.1)	100 (0.2)	3156 (10.5)	91 (0.0)
MIXFeed	0	0	232808 (78.7)	30688 (75.8)	18363 (61.3)	183757 (81.6)
	1		62929 (21.3)	9797 (24.2)	11605 (38.7)	41527 (18.4)
CortisoneMother	0.0	200947	85832 (90.5)	2825 (94.6)	7342 (66.3)	75665 (93.7)
	1.0		8958 (9.5)	160 (5.4)	3729 (33.7)	5069 (6.3)
AntibioticsBaby	0.0	1	274744 (92.9)	38229 (94.4)	18841 (62.9)	217674 (96.6)
	1.0		20992 (7.1)	2256 (5.6)	11127 (37.1)	7609 (3.4)
AnswerIX	1	0	292247 (98.8)	39835 (98.4)	29603 (98.8)	222809 (98.9)
	2		3450 (1.2)	635 (1.6)	358 (1.2)	2457 (1.1)
	3		39 (0.0)	15 (0.0)	6 (0.0)	18 (0.0)
	4		1 (0.0)	///	1 (0.0)	///
Premature	0.0	50411	229642 (93.6)	36603 (97.8)	9195 (55.0)	183844 (96.2)
	1.0		15684 (6.4)	828 (2.2)	7531 (45.0)	7325 (3.8)
Sampling	Basale già noto	0	177 (0.1)	13 (0.0)	66 (0.2)	98 (0.0)
	Controllo		7012 (2.4)	665 (1.6)	2155 (7.2)	4192 (1.9)
	Iniziale		288540 (97.6)	39807 (98.3)	27747 (92.6)	220986 (98.1)
	BIS		8 (0.0)	///	///	8 (0.0)
BIS	0.0	237374	9066 (15.5)	118 (6.7)	1381 (9.6)	7567 (17.9)
	1.0		49297 (84.5)	1636 (93.3)	13013 (90.4)	34648 (82.1)
ENFeed	0	0	295259 (99.8)	40480 (100.0)	29657 (99.0)	225122 (99.9)
	1		478 (0.2)	5 (0.0)	311 (1.0)	162 (0.1)
TPNMCTFeed	0	0	295706 (100.0)	40485 (100.0)	29945 (99.9)	225276 (100.0)
	1		31 (0.0)	///	23 (0.1)	8 (0.0)
Sex	F	0	143363 (48.5)	19688 (48.6)	13703 (45.7)	109972 (48.8)
	M		152374 (51.5)	20797 (51.4)	16265 (54.3)	115312 (51.2)
Etnia	Arab	2	17642 (6.0)	2097 (5.2)	1835 (6.1)	13710 (6.1)
	Asian		13044 (4.4)	1118 (2.8)	1463 (4.9)	10463 (4.6)

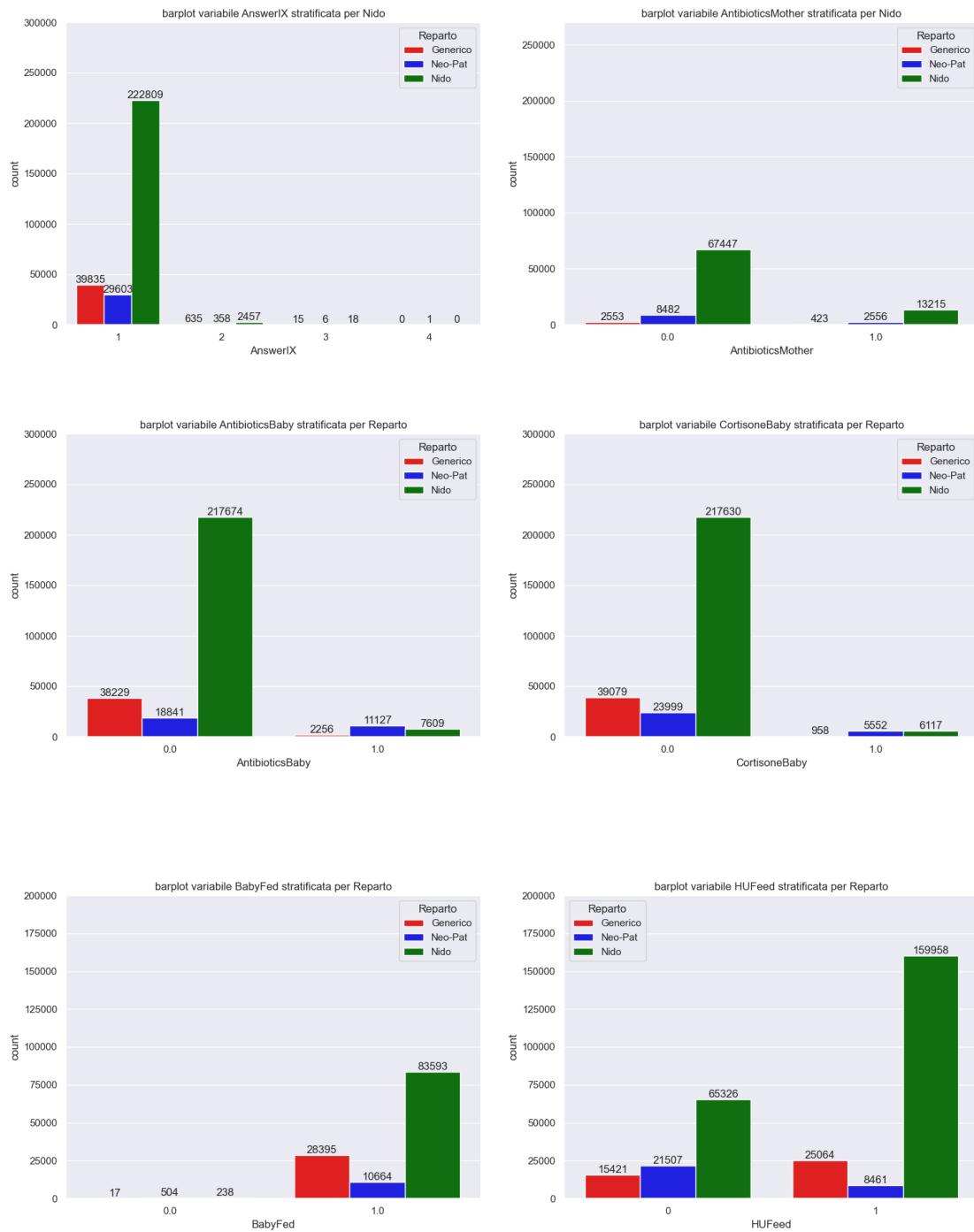
	Caucasian		244921 (82.8)	35552 (87.8)	23989 (80.1)	185380 (82.3)
	Hispanic/Latino		7191 (2.4)	216 (0.5)	762 (2.5)	6213 (2.8)
	Native Hawaiian or Other Pacific Islander		12745 (4.3)	1502 (3.7)	1896 (6.3)	9347 (4.1)
	Black or African American		183 (0.1)	///	19 (0.1)	164 (0.1)
	Other		9 (0.0)	///	3 (0.0)	6 (0.0)
ARTFeed	0	0	267521 (90.5)	37385 (92.3)	22762 (76.0)	207374 (92.1)
	1		28216 (9.5)	3100 (7.7)	7206 (24.0)	17910 (7.9)
TyroidMother	0.0	2151	250201 (85.2)	37112 (92.5)	26266 (88.8)	186823 (83.5)
	1.0		43385 (14.8)	3021 (7.5)	3314 (11.2)	37050 (16.5)
HUFeed	0	0	102254 (34.6)	15421 (38.1)	21507 (71.8)	65326 (29.0)
	1		193483 (65.4)	25064 (61.9)	8461 (28.2)	159958 (71.0)
Meconium	0.0	2103	293178 (99.8)	40038 (99.9)	29303 (99.1)	223837 (99.9)
	1.0		456 (0.2)	51 (0.1)	273 (0.9)	132 (0.1)
CortisoneBaby	0.0	2402	280708 (95.7)	39079 (97.6)	23999 (81.2)	217630 (97.3)
	1.0		12627 (4.3)	958 (2.4)	5552 (18.8)	6117 (2.7)
BirthMethod	Altro	53462	8856 (3.7)	476 (4.4)	570 (2.2)	7810 (3.8)
	Cesareo		74697 (30.8)	3842 (35.5)	14920 (58.1)	55935 (27.2)
	Naturale		158722 (65.5)	6513 (60.1)	10194 (39.7)	142015 (69.0)

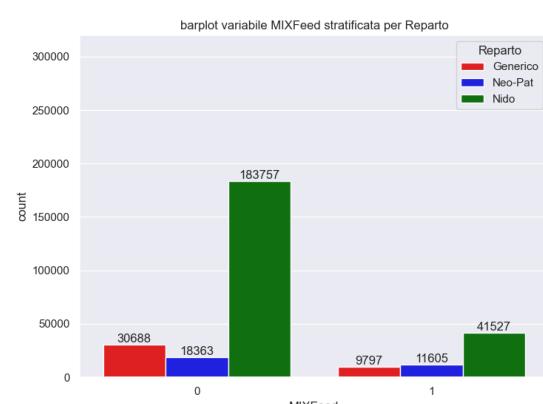
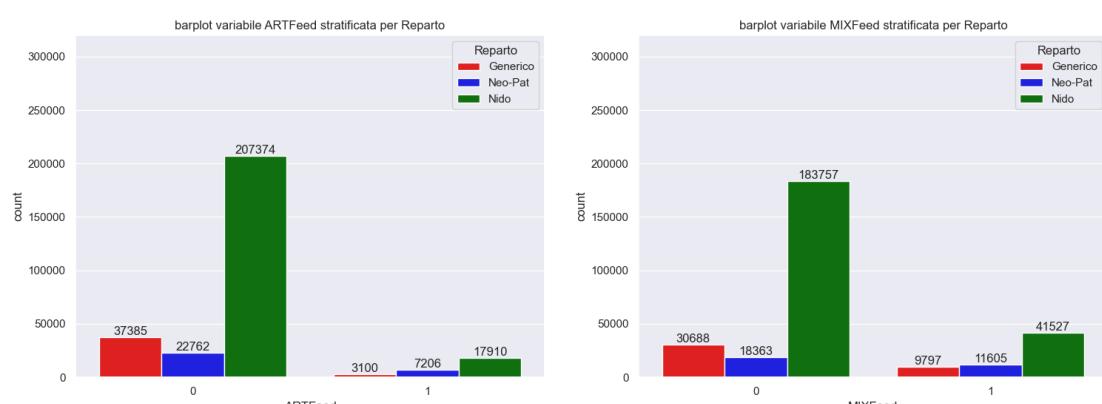
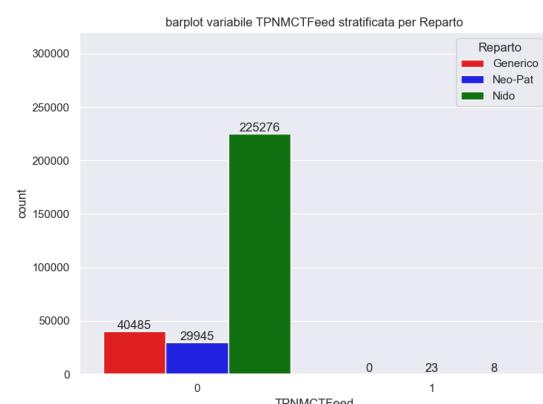
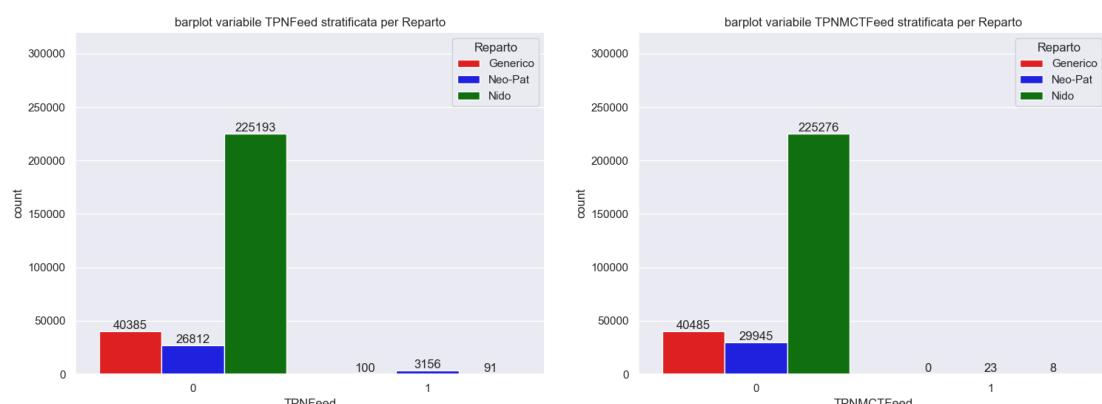
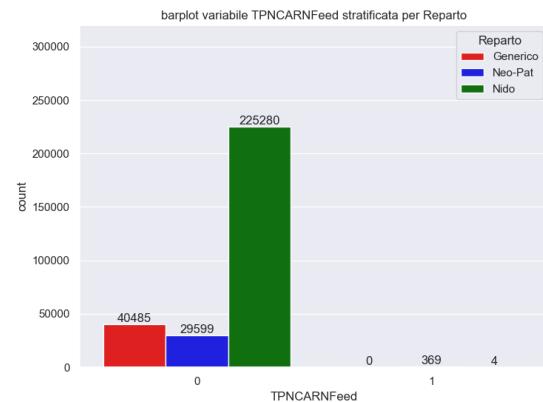
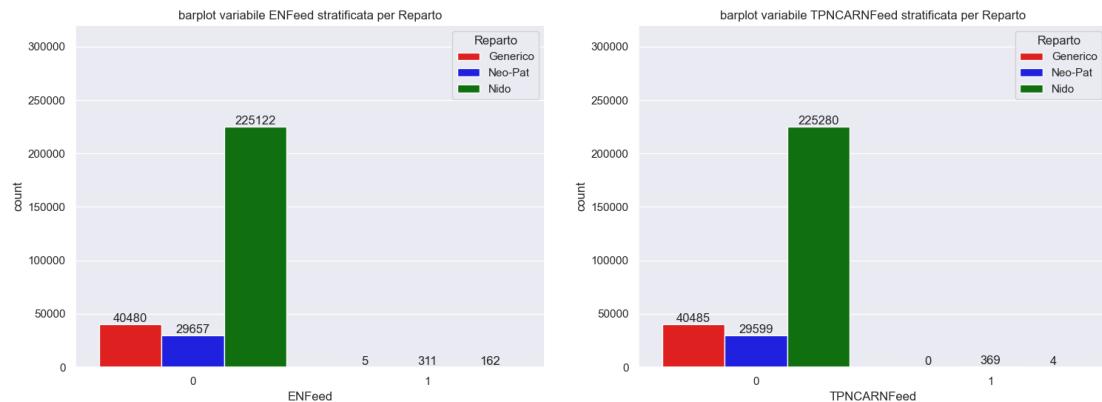
Table 2.11: distribuzioni variabili qualitative stratificate per variabile Reparto

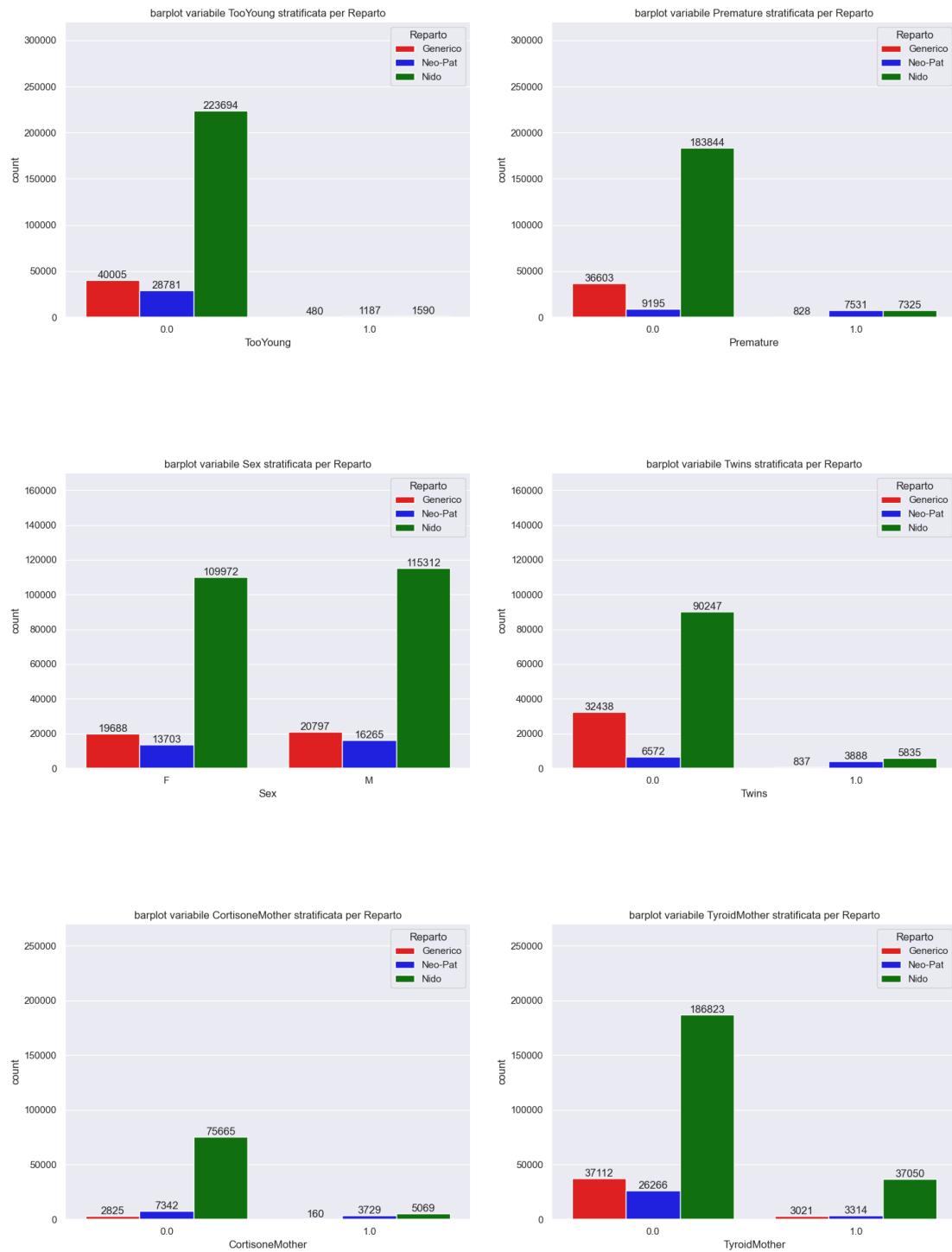
Da una prima osservazione dei risultati ottenuti [2.11], è facile notare che alcune distribuzioni delle variabili qualitative non sembrano condizionate dalla variabile reparto, con proporzioni simili per ogni categoria (come le variabili *Meconium* o *AnswerIX*), mentre altre sono condizionate dalle differenze di reparto (nel reparto neo-patologico aumenta sensibilmente la percentuale di parti gemellari, cesarei e prematuri, cambiano le abitudini di alimentazione dei neonati e in generale ci sono differenze rispetto alle distribuzioni legate agli altri due reparti). Come nel caso dell'intero dataset, è opportuno indagare ulteriormente queste distribuzioni con grafici ad hoc.

2.3.4 Barplot variabili qualitative

Osserviamo ora i barplot delle variabili qualitative stratificate per la variabile *Reparto* (in rosso le distribuzioni per *Reparto Generico*, in blu le distribuzioni per *Reparto Neo-patologico*, in verde le distribuzioni per *Reparto Nido*).







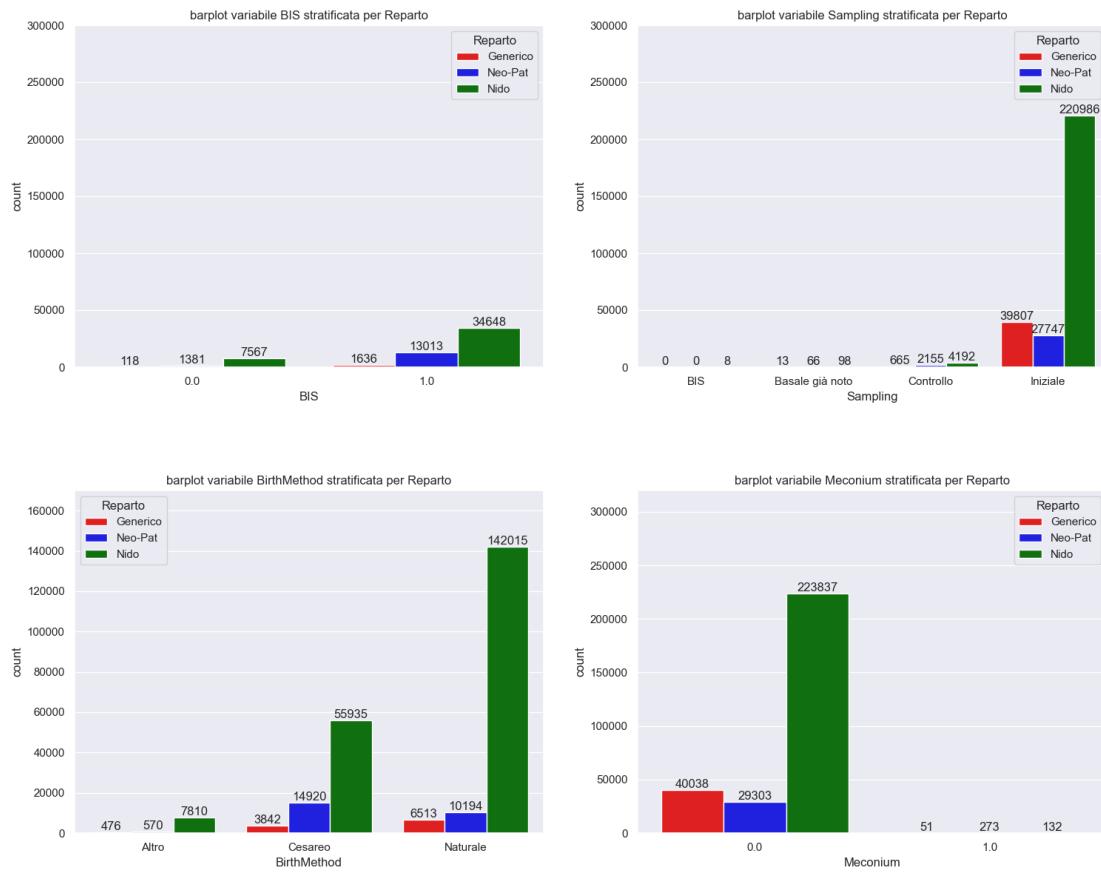


Figure 2.8: barplot variabili qualitative stratificate per Reparto

Alcune delle differenze riportate nella tabella delle distribuzioni stratificate [2.11] sono subito chiare con una visualizzazione grafica, come le differenze di proporzioni tra parti gemellari e parti prematuri tra i diversi reparti.

Ovviamente in numeri assoluti sono sempre molto più numerosi i casi legati a reparti “Nido” o “Generico”, dunque di solito privi di complicanze particolari, ma per la natura del dataset e lo scopo del progetto è sempre bene considerare le caratteristiche anche di pochi parti con complicanze, in quanto per molte malattie metaboliche l’individuazione preventiva e l’assunzione di cure fin dalla nascita possono portare benefici importanti e duraturi.

3. La riduzione della dimensionalità

3.1.1 Introduzione

La riduzione di dimensionalità [\[Banks 04\]](#) [\[Lee 07\]](#) è uno dei passaggi fondamentali dell'analisi di dati e dell'apprendimento automatico, e si concentra sulla riduzione del numero di variabili, in dataset solitamente di grandi dimensioni, mantenendo intatte le informazioni essenziali e riducendo al minimo la perdita di informazioni generale.

3.1.2 Cenni storici

Nella storia dell'analisi dei dati, i primi esempi di riduzione di dimensionalità [\[Maaten 09\]](#) [\[Ghodsi 06\]](#) possono essere ricondotti ai primi anni del XX secolo, con contributi significativi in svariati ambiti; la tecnica ha poi continuato a guadagnare importanza nel corso del '900, anche come conseguenza dell'aumentare sia delle capacità computazionali a disposizione degli studiosi, sia delle necessità e dei possibili ambiti di applicazione delle metodologie.

Tra le tecniche più antiche si trova la PCA, sviluppata nel 1901 da Karl Pearson, che ha gettato le basi per la riduzione di dimensionalità nell'ambito delle statistiche multivariate, mirando a trasformare una serie di variabili correlate in poche variabili non correlate, note come componenti principali. Negli anni '30 altre tecniche vengono teorizzate ed approfondite, a partire dal Multidimensional Scaling (MDS) introdotto da Torgny Segerstedt, che mirava a rappresentare distanze o dissimilarità a coppie tra i punti dati in uno spazio a dimensioni inferiori, con applicazioni dalla psicologia alla geografia.

Alcune delle tecniche in analisi in questo lavoro sono state introdotte abbastanza di recente [\[Reddy 20\]](#), come la t-SNE (teorizzata tra il 2010 e il 2015 da scienziati quali Zhirong Yang, Miguel A Carreira-Perpinan e soprattutto Laurens van der Maaten) e l'UMAP (frutto del lavoro di Leland McInnes, John Healy e James Melville intorno al 2018/2020).

3.1.3 Obiettivi della riduzione di dimensionalità

La riduzione di dimensionalità ha una serie di scopi specifici, tipici dell'analisi dei dati e dell'apprendimento automatico:

- **Risolvere il “Curse of dimensionality”:** il Curse of dimensionality è un problema sempre più ricorrente negli ultimi anni, in contesti nati col Web 3.0 dove la quantità di dati a disposizione cresce a dismisura in pochissimo tempo; la necessità di analizzare i dati, e creare valore a partire da questi, ha portato gli esperti a dover analizzare dati a dimensionalità molto elevata, con conseguente aumento della complessità computazionale, difficoltà nel ricavare risultati validi per tutti i dati e problemi come l'overfitting o la presenza di outliers. La riduzione di dimensionalità permette di mitigare questi problemi e di ottenere comunque risultati validi e significativi.
- **Necessità di efficienza computazionale:** semplificare il dataset a disposizione, riducendone la dimensionalità, permette di velocizzarne gli algoritmi, in modo da essere particolarmente vantaggioso per compiti strettamente legati alla riduzione di dimensionalità, come la cluster analysis, algoritmi di classificazione e di regressione.
- **Tecniche di visualizzazione:** spesso alcuni problemi legati a cluster analysis, classificazione e regressione derivano dall'impossibilità di ottenere visualizzazioni dei risultati ottenuti, data l'alta dimensionalità dei dati; ottenere rappresentazioni a dimensioni inferiori permette non solo una visualizzazione efficace dei risultati ottenuti, ma anche di interpretare più semplicemente modelli e strutture dei dati.

- **Riduzione del rumore:** concentrandosi su alcune delle caratteristiche più rilevanti dei dati, la riduzione di dimensionalità può contribuire attivamente ad eliminare rumore ed informazioni non pertinenti dai dati stessi, migliorando la generalizzazione del modello.

Rimuovendo le caratteristiche non rilevanti, la riduzione di dimensionalità può migliorare la capacità di generalizzazione dei modelli di apprendimento automatico. Tuttavia, allo stesso tempo, la riduzione di dimensionalità può comportare una certa perdita di informazioni, influenzando potenzialmente l'accuratezza dei modelli, a maggior ragione considerando che le prestazioni di alcuni metodi di riduzione di dimensionalità possono essere particolarmente sensibili alla scelta dei parametri o alle caratteristiche dei dati.

Inoltre, l'efficacia dei metodi di riduzione di dimensionalità dipende particolarmente dalla scelta della metodologia applicata in base alle caratteristiche specifiche del dataset (e all'interno della stessa tecnica, dipende fortemente dal tuning dei parametri), e non esiste un metodo singolo universalmente adatto a tutte le situazioni, contesti e dati.

3.1.4 Parametri chiave nella riduzione di dimensionalità

Le tecniche di riduzione della dimensionalità sviluppate negli anni sono molteplici, e non è mai semplice andare ad individuare la tecnica più adatta in ogni contesto, a livello di ambito di applicazione (alcune tecniche risultano più adatte per certe tipologie di dati), come formato dei dati a disposizione (alcune tecniche lavorano meglio con dimensioni e quantità di dati differenti, e con tipologie di dati a disposizione molto diverse) e come scelta dei parametri in base alla tecnica scelta. Tuttavia, alcuni parametri sono comuni ad una moltitudine di tecniche di riduzione di dimensionalità, tra cui:

- **Numero di componenti (k):** questo parametro determina il numero di componenti principali; si tratta senza dubbio del parametro più importante di alcune tecniche di riduzione della dimensionalità, come per la PCA (Principal Component Analysis). La scelta appropriata di k è cruciale per bilanciare la conservazione delle informazioni e la riduzione di dimensionalità; al contrario, se il focus è sulla visualizzazione, a costo di perdere informazioni la scelta può ricadere su due (o al massimo tre) dimensioni, in modo da ottenere spazi bi/tridimensionali.
- **Quantità di varianza residua:** in alcune tecniche, come la PCA, è possibile specificare la proporzione di varianza da mantenere, guidando la selezione delle componenti principali; con la scelta di questo parametro si riesce a raggiungere un giusto compromesso tra riduzione di dimensionalità e minor perdita possibile di informazioni.
- **Dimensione del vicinato:** alcune metodologie come Isomap richiedono di impostare il numero di vicini da considerare quando si stimano le distanze geodetiche; questo parametro dunque influenza direttamente sulla precisione della rappresentazione della varietà.
- **Tasso di apprendimento:** il tasso di apprendimento è un parametro che influenza il processo di ottimizzazione in alcuni algoritmi, tra cui t-SNE; per questo motivo, è necessaria la sperimentazione con diversi tassi di apprendimento, per evitare problemi di convergenza.

Ogni tecnica di riduzione di dimensionalità ha una serie di parametri caratteristica della metodologia stessa, ognuno con criteri di scelta ed indicazioni ottimali da seguire per la scelta dei valori e dei metodi. Infine, è fondamentale fare attenzione ad alcuni aspetti legati alla natura stessa dei dati di interesse, che possono condizionare in maniera significativa i risultati ottenuti, oltre ai tempi di esecuzione delle metodologie di dimensionality reduction:

- **Alta dimensionalità:** ovviamente la riduzione di dimensionalità è particolarmente indicata e vantaggiosa quando bisogna trattare dataset ad alta dimensionalità, in cui il numero di caratteristiche è molto grande (in alcuni casi addirittura più alto del numero di campioni a disposizione).

-
- **Struttura intrinseca:** i dati dovrebbero mostrare una struttura intrinseca, o modelli per garantire una riduzione di dimensionalità significativa; se i dati mancano di struttura, la riduzione di dimensionalità potrebbe non produrre risultati informativi.
 - **Tipologia di dati:** le tecniche di riduzione di dimensionalità possono essere applicate a svariate tipologie di dati, inclusi dati quantitativi, qualitativi o di tipo misto; la scelta della tecnica dipende spesso anche dal tipo di dati a disposizione.
 - **Qualità dei dati:** dati di alta qualità, privi di rumore e valori mancanti, e con pochi o nessun outlier (motivo per cui è sempre consigliata una fase di normalizzazione/standardizzazione dei dati) migliorano sensibilmente l'efficacia di tutte le tecniche di riduzione di dimensionalità; per questi motivi, la fase di pulizia dei dati è sempre fondamentale per ottenere risultati significativi.

3.1.5 Applicazioni della riduzione di dimensionalità

Le tecniche di riduzione della dimensionalità sono nate ad inizio XX secolo, di pari passo con altri step fondamentali nel processo di analisi dei dati (come tecniche di cluster analysis, ad esempio), a partire dalla necessità di ottenere dati puliti, chiari e significativi in ottica di applicazione di altre tecniche quali regressione, classificazione e clusterizzazione. Si tratta ovviamente di metodologie con fine applicativo, dunque è facile immaginare che gli ambiti di ricerca si siano ampliati enormemente col passare del tempo. Tra gli utilizzi più comuni delle tecniche di riduzione della dimensionalità troviamo:

- **Elaborazione di immagini e video:** alcune tecniche di riduzione di dimensionalità vengono spesso impiegate in sistemi atti al riconoscimento facciale, il rilevamento di oggetti, la compressione spaziale di files (riducendo al minimo la perdita di informazioni, dunque anche di qualità in caso di foto e video) e la sintesi video; in questi processi, tecniche come la PCA e la t-SNE vengono impiegate per ottenere procedure più rapide, efficaci e computazionalmente più leggere per i sistemi.
- **Riconoscimento del parlato:** la riduzione della dimensionalità delle caratteristiche audio migliora l'efficienza dei sistemi di riconoscimento del parlato, permettendo di ottenere risultati più rapidi ed efficienti nonostante quantità di dati molto elevate, andando a riconoscere features fondamentali nel riconoscimento vocale, e procedendo con tecniche che aiutano le procedure (come la noise reduction o il riconoscimento della lingua parlata).
- **Text Mining:** la riduzione di dimensionalità facilita l'estrazione di caratteristiche significative da grandi corpus di dati testuali, agevolando compiti come la sentiment analysis, l'information retrieval (ridurre le informazioni a quelle fondamentali per indirizzare ottimamente le soluzioni delle ricerche), il topic modeling, la classificazione di testi, i sistemi di raccomandazione e il clustering dei documenti.
- **Finanza:** l'ottimizzazione del portafoglio, le previsioni di andamenti finanziari, il rilevamento delle frodi e la valutazione del rischio traggono tutte beneficio da tecniche di riduzione di dimensionalità applicate nell'analisi di dati finanziari.
- **Scienze ambientali:** nelle ricerche climatiche ed ambientali, tecniche di riduzione di dimensionalità aiutano a comprendere set di dati complessi, come le immagini satellitari o le progettazioni di complessi modelli climatici.
- **Robotica:** l'estrazione di caratteristiche attraverso la riduzione di dimensionalità migliora l'efficienza dei sistemi robotici semplificando la fase di raccolta e processamento di dati dei sensori, aiutando nella creazione, nell'ottimizzazione e nel design di circuiti analogici.
- **Rilevamento delle anomalie:** tecniche di riduzione della dimensionalità possono essere impiegate, insieme a metodologie di cluster analysis ad esempio, col fine di identificare outliers, modelli insoliti e tentativi di frode in vari settori, tra cui sicurezza delle reti, controllo della qualità industriale e monitoraggio dei processi produttivi industriali.

3.1.6 Focus sulle applicazioni mediche e biostatistiche

Negli ultimi anni, ci sono stati sempre più sviluppi e utilizzi di tecniche di riduzione della dimensionalità in contesto medico e clinico, con ambiti di applicazione quali:

- **Analisi di immagini mediche:** la riduzione di dimensionalità è cruciale nell'elaborazione e interpretazione di immagini mediche, come risonanze magnetiche o tomografie computerizzate, trattandosi di immagini ad alta definizione; la riduzione di dimensionalità è un passaggio cruciale in studi clinici sull'imaging medico, facilitando diagnosi, semplificando la creazione di modelli predittivi e contribuendo alla pianificazione di trattamenti mirati.
- **Scoperta ed analisi di farmaci:** nella ricerca farmaceutica, la riduzione di dimensionalità aiuta nell'analisi di descrittori molecolari e prodotti chimici, e nell'identificazione di potenziali candidati farmacologici, rivelando relazioni struttura-attività.
- **Cartelle cliniche elettroniche (EHR):** l'analisi delle cartelle cliniche dei pazienti è un'attività sempre più automatizzata, che coinvolge tecniche di riduzione della dimensionalità per scoprire patterns nascosti, supportare la decisione clinica e migliorare la gestione sanitaria; come in altri casi, la dimensionality reduction può essere usata in combinazione con altre metodologie, come la cluster analysis, per scoprire clusters di pazienti con profili clinici simili e preparare terapie e cure ad hoc.
- **Diagnosi e prognosi delle malattie:** la riduzione di dimensionalità può essere molto utile nell'identificare caratteristiche rilevanti per la classificazione delle malattie e nella previsione degli esiti di cure su particolari clusters di pazienti, il tutto basato su grandi quantità di dati clinici o genetici.
- **Studi epidemiologici:** nella ricerca epidemiologica, la riduzione di dimensionalità migliora la comprensione della diffusione delle malattie e della dinamica di trasmissione, contribuendo all'identificazione di patterns e fattori di rischio, ed indirizzando su caratteristiche più o meno ricorrenti nei pazienti malati.
- **Stratificazione dei pazienti:** nella medicina di precisione, la riduzione di dimensionalità è fondamentale nell'aiutare a stratificare i pazienti in base a profili molecolari o clinici, guidando strategie di trattamento personalizzato.
- **Identificazione di biomarcatori:** l'analisi di dati omici ad alta dimensionalità, come genomica o proteomica, coinvolge in primo piano tecniche di riduzione della dimensionalità per identificare biomarcatori associati alle malattie.
- **Studi di neuroimaging e malattie mentali:** nelle neuroscienze, le tecniche di riduzione di dimensionalità vengono applicate a dati di imaging funzionale e strutturale del cervello, aiutando a comprendere la funzione cerebrale e a identificare reti neurali; inoltre, si possono identificare sottotipi di disturbi mentali grazie all'analisi di dati psicologici e comportamentali.
- **Clustering in biostatistica:** la riduzione di dimensionalità è spesso strettamente collegata a tecniche di cluster analysis, col fine di scoprire patterns e sottopopolazioni all'interno di dataset biologici, migliorando la comprensione dell'eterogeneità delle malattie.

3.1.7 Integrazione con tecniche di cluster analysis

La riduzione di dimensionalità e la cluster analysis sono spesso interconnesse [Hozumi 21], con la riduzione di dimensionalità che funge da step di pre-elaborazione per le tecniche di cluster analysis, ed aiuta nell'interpretazione dei risultati del clustering grazie alla possibilità di visualizzare i risultati con grafici bidimensionali. Le due metodologie sono strettamente legate per diversi motivi:

- **Miglioramento della visualizzazione:** le rappresentazioni a dimensioni ridotte ottenute a partire dalla riduzione di dimensionalità facilitano la visualizzazione dei clusters, rendendo molto più semplice l'interpretazione delle strutture complesse ottenute con il clustering.

-
- **Miglioramento delle prestazioni di tecniche di clustering:** concentrandosi sulle caratteristiche rilevanti, la riduzione di dimensionalità permette di migliorare le prestazioni del clustering, specialmente a partire da spazi ad alta dimensionalità.
 - **Identificazione di sottopopolazioni nei dati:** tecniche di riduzione di dimensionalità sono impiegate per scoprire la presenza di sottopopolazioni nascoste nei dati, fornendo informazioni sulla diversità dei clusters.
 - **Estrazione di caratteristiche per il clustering:** il dataset ridotto, ottenuto a partire dalle caratteristiche considerate fondamentali e più esplicative dalle tecniche di riduzione della dimensionalità, è un input fondamentale per algoritmi di clustering, andando a migliorare l'efficienza e l'interpretabilità dei dati a disposizione.

3.2 Principal Component Analysis (PCA)

3.2.1 Introduzione

La Principal Component Analysis (PCA) [Lee 07] è una tecnica fondamentale nell'analisi multivariata e nella riduzione della dimensionalità, ampiamente utilizzata in svariati settori, quali la statistica, l'apprendimento automatico e l'elaborazione di segnali. La PCA è un metodo che mira a trasformare dati ad alta dimensionalità in una rappresentazione a dimensionalità inferiore, preservando il più possibile della varianza originale. Questa analisi completa fornirà una comprensione dettagliata della PCA, esplorando il suo scopo, i parametri, i meccanismi di regolazione, le applicazioni, i passaggi algoritmici e valutando i suoi pro, i suoi contro e i risultati ottenuti.

Lo scopo principale della PCA è ridurre la dimensionalità di un dataset identificando le direzioni, chiamate componenti principali, lungo le quali i dati variano di più. L'idea è proiettare i dati originali su un nuovo sistema di coordinate definito da queste componenti principali, catturando così le caratteristiche essenziali dei dati ed eliminando le informazioni ridondanti. La PCA è ampiamente utilizzata per la visualizzazione dei dati, la riduzione del rumore e l'estrazione delle caratteristiche.

La PCA realizza la riduzione della dimensionalità trasformando i dati in un insieme di variabili non correlate, le componenti principali, ordinate per la quantità di varianza che spiegano. Le prime componenti principali conservano la maggior parte delle informazioni, consentendo una compressione efficace dei dati con una perdita minima di informazioni. Infatti in questo metodo di riduzione della dimensionalità è cruciale la quantità di varianza delle componenti principali: maggiore varianza indica una rappresentazione più fedele dei dati originali.

La PCA trova applicazioni in diversi settori [Qureshi 17] [Kurita 20] grazie alla sua versatilità nella gestione di dati ad alta dimensionalità. Ecco cinque esempi significativi:

- **Compressione di immagini:** la PCA è fondamentale per la riduzione dello spazio di archiviazione richiesto per le immagini digitali; la Principal Component Analysis può essere applicata a set di dati di immagini per catturare le caratteristiche più significative, consentendo una compressione sostanziale pur mantenendo le informazioni visive essenziali.
- **Riconoscimento facciale in Computer Vision:** per l'identificazione di volti in immagini o video e l'estrazione delle componenti principali delle caratteristiche facciali, l'Analisi delle Componenti Principali è un metodo tra i più utilizzati; infatti, la PCA riduce la dimensionalità dei dati facciali conservando informazioni cruciali per un riconoscimento rapido ed accurato.
- **Analisi dei dati genomici in bioinformatica:** la PCA è una tecnica particolarmente indicata per l'analisi dei dati di espressione genica, aiutando ad identificare patterns e relazioni nei dati genomici, conservando informazioni sulle dinamiche biologiche sottostanti.
- **Elaborazione del segnale vocale:** la PCA è utile per l'estrazione di caratteristiche essenziali dai segnali audio; viene infatti applicata per ridurre la dimensionalità dei dati di segnali vocali, catturando le caratteristiche acustiche più significative per compiti come il riconoscimento della lingua o la rilevazione delle emozioni.
- **Finanza e gestione di portafogli:** la PCA è impiegata per l'analisi e la gestione di portafogli finanziari, per identificare le componenti principali dei rendimenti degli asset finanziari e per aiutare nella costruzione di un portafoglio diversificato catturando le principali fonti di variazione nei prezzi degli asset.

3.2.2 Iperparametri e tuning

La PCA è una tecnica con pochi parametri nella sua forma di base, ma ci sono alcune considerazioni da fare legate ai pochi parametri del metodo, in primis la scelta del numero di componenti principali da mantenere:

- **n_components (default: ‘None’)**: si tratta del numero di componenti principali da mantenere; può essere specificato in diversi modi (col default ‘None’ vengono conservate tutte le componenti, con un valore interno viene mantenuto un numero di componenti dati, con un valore decimale compreso tra 0 e 1 viene conservata una quantità di componenti tale da avere la quantità di varianza spiegata superiore alla percentuale di componenti).
- **whiten (default: ‘False’)**: il whitening rimuove alcune informazioni dal segnale trasformato, ma contemporaneamente può aiutare nella precisione delle previsioni degli stimatori.

3.2.3 Passaggi algoritmici

La PCA segue una serie di passaggi [\[Karamizadeh 13\]](#) [\[Kurita 20\]](#) per trasformare dati ad alta dimensionalità in una rappresentazione a dimensionalità inferiore:

1. **Normalizzazione/standardizzazione e pulizia dei dati**: si tratta di un passaggio fondamentale per assicurare che tutte le caratteristiche contribuiscano in modo equo alla varianza: infatti, outliers e dati su scale di misura differenti possono condizionare sensibilmente i risultati ottenuti.
2. **Calcolo della matrice di covarianza**: viene calcolata la matrice di covarianza dei dati standardizzati (matrice simmetrica, con le varianze sulla diagonale principale) che rappresenta le relazioni tra diverse caratteristiche, indicando come variano congiuntamente (osservando dunque non solo le singole variazioni all'interno dei dati, ma anche se queste sono correlate).
3. **Autodecomposizione**: eseguire l'autodecomposizione (detta anche eigendecomposition) della matrice di covarianza per ottenere i suoi autovalori e gli autovettori corrispondenti. Gli autovalori rappresentano la varianza spiegata da ciascuna componente principale, e gli autovettori definiscono le direzioni di queste componenti.
4. **Selezione delle Componenti Principali**: in questa fase vengono ordinati gli autovalori in ordine decrescente, in modo da scegliere i primi k autovettori corrispondenti ai k autovalori più grandi (e dunque alle k dimensioni considerate); gli autovettori trovati formano le componenti principali.
5. **Proiezione**: come passaggio finale, vengono proiettati i dati originali sul sottospazio definito dalle componenti principali selezionate, in modo da ottenere una rappresentazione a dimensionalità inferiore rispetto ai dati iniziali.

3.2.4 Vantaggi e svantaggi

L'Analisi delle Componenti Principali presenta una serie di vantaggi [\[Karamizadeh 13\]](#), tra cui:

- **Riduzione della dimensionalità**: la PCA è fondamentale per ridurre efficacemente la dimensionalità dei dati, rendendoli più gestibili e leggibili, migliorando l'efficienza computazionale e riducendo drasticamente i tempi di esecuzione di algoritmi successivi alla PCA.
- **Estrazione delle features**: le componenti principali catturano le variazioni più significative nei dati, risolvendo il problema della multicollinearità e fornendo una rappresentazione concisa della struttura del dataset.
- **Visualizzazione**: l'utilizzo di tecniche di riduzione della dimensionalità, come la PCA, permettono la creazione di visualizzazioni, facilitando l'esplorazione e l'interpretazione dei dati a disposizione.
- **Indipendenza lineare**: le componenti principali ottenute dalla PCA sono linearmente indipendenti, semplificando le analisi e la modellazione successive.

Tuttavia, l'algoritmo presenta anche alcune criticità, tra cui:

- **Perdita di informazioni:** nonostante la conservazione della maggior parte della varianza, la PCA comporta inevitabilmente una perdita di informazioni, specialmente quando si utilizza un numero particolarmente ridotto di componenti principali (come due o tre componenti, quando c'è necessità di ottenere visualizzazioni grafiche).
- **Sensibilità agli outliers:** la PCA è particolarmente sensibile agli outliers, poiché possono influenzare in modo sproporzionato il calcolo delle componenti principali (motivo per cui la fase di standardizzazione è cruciale per ottenere risultati validi).
- **Interpretabilità:** nonostante la riduzione di dimensionalità fornisca una rappresentazione a dimensionalità ridotta, l'interpretazione del significato delle singole componenti principali potrebbe non essere immediata in alcuni casi.
- **Applicabilità a certe distribuzioni di dati:** la PCA è particolarmente efficace per dati distribuiti secondo una Gaussiana, e le sue prestazioni possono peggiorare sensibilmente per distribuzioni non gaussiane.

3.3 t-distributed Stochastic Neighbor Embedding (t-SNE)

3.3.1 Introduzione

La t-distributed Stochastic Neighbor Embedding (t-SNE) è una potente tecnica di riduzione della dimensionalità, ampiamente utilizzata nell'apprendimento automatico e nella visualizzazione dei dati. Sviluppata da Laurens van der Maaten e Geoffrey Hinton, la t-SNE è un metodo non lineare di riduzione della dimensionalità particolarmente efficace nel rivelare la struttura intrinseca dei dati ad alta dimensionalità in uno spazio a dimensionalità inferiore.

Il principale scopo della t-SNE è la visualizzazione e l'esplorazione della struttura sottostante dei dati ad alta dimensionalità, mappandoli in uno spazio a dimensionalità inferiore e preservando al contempo le similarità tra coppie di punti dati. A differenza delle tecniche lineari come l'Analisi delle Componenti Principali (PCA), la t-SNE eccelle nel catturare relazioni complesse e non lineari nei dati (il focus infatti, rispetto a tecniche come la PCA, è di preservare la similarità tra vicini piuttosto che la variabilità dei dati). È particolarmente utile per visualizzare clusters, e strumento essenziale per l'analisi esplorativa dei dati e il riconoscimento di modelli.

La t-SNE raggiunge questi obiettivi costruendo una distribuzione di probabilità su coppie di punti dati ad alta dimensionalità ed una distribuzione simile su punti corrispondenti nello spazio a dimensionalità ridotta. L'algoritmo minimizza la divergenza tra queste due distribuzioni, avvicinando efficacemente punti simili nella rappresentazione a dimensionalità inferiore.

La t-SNE trova applicazioni [\[Maaten 08\]](#) in vari settori grazie alla sua capacità di catturare strutture complesse e rivelare relazioni intricate nei dati ad alta dimensionalità:

- **Dati di sequenziamento di RNA a cellula singola in biologia:** t-SNE viene utilizzata per l'analisi dei profili di espressione genica e l'interpretazione a livello di singola cellula, in modo da identificare popolazioni cellulari distinte e comprendere le loro relazioni basate sui modelli di espressione genica.
- **Incorporamento di caratteristiche di immagini per Computer Vision:** la riduzione di vettori di caratteristiche di immagini ad alta dimensionalità in uno spazio a dimensionalità inferiore è utile per l'analisi, aiutando a visualizzare e comprendere le relazioni tra le immagini basate sulle loro caratteristiche.
- **Elaborazione del Natural Language:** il metodo t-SNE viene utilizzato per l'analisi e le rappresentazioni di corpus di documenti, in modo da visualizzare ed esplorare relazioni semantiche tra parole o documenti, per rivelare clusters di parole o documenti semanticamente simili.
- **Rilevamento di anomalie in sicurezza informatica:** la t-SNE può essere applicata per visualizzare dati di traffico di rete e identificare clusters di comportamenti anomali, che rappresentano potenziali minacce alla sicurezza.
- **Scoperta di farmaci in chimica:** è un metodo utile per esplorare e visualizzare le relazioni tra composti chimici, ed aiuta ad identificare clusters di composti con proprietà simili, guidando i ricercatori nella scoperta di nuovi farmaci efficaci e sicuri.

3.3.2 Iperparametri e tuning

La t-SNE coinvolge alcuni parametri chiave [\[Cai 22\]](#) [\[Cao 17\]](#) [\[distill.pub\]](#), e la loro regolazione appropriata è cruciale per ottenere visualizzazioni significative (a differenza di metodi come la PCA, ha molti parametri da definire). Il parametro principale è la perplessità, ma è fondamentale porre particolare attenzione anche ai valori di altri parametri molto importanti per il metodo:

- **perplexity (default: 30.0)**: iperparametro che influenza l'equilibrio tra la conservazione delle strutture globali e locali nei dati (può essere considerata come una misura del numero effettivo di vicini per ciascun punto dati); a valori più alti corrispondono strutture globali più enfatizzate (e sono solitamente associati a dataset ad alta dimensionalità), viceversa per valori più bassi viene data maggiore importanza alle strutture locali; è importante non superare mai il numero di campioni nei dati in studio col valore della perplexity.
- **n_components (default: 2)**: si tratta del numero di componenti principali, quindi anche delle dimensioni dello spazio ridotto ottenuto dalla procedura (come nei casi degli altri metodi di riduzione di dimensionalità, valori pari a 2 o 3 permettono di osservare graficamente i risultati ottenuti).
- **early_exaggeration (default: 12.0)**: questo parametro definisce quanto i campioni nei clusters vengono riprodotti vicini nello spazio di dimensionalità ridotto; a valori elevati corrispondono clusters più sparsi e meno compatti internamente; bisogna porre comunque particolare attenzione a questo parametro, nonostante non incida significativamente sui risultati ottenuti, poiché può comunque aumentare esponenzialmente i tempi di esecuzione dell'algoritmo con valori troppo elevati per i dati in studio.
- **learning_rate (default: ‘auto’)**: il *learning_rate* è un parametro che definisce in maniera significativa la forma della distribuzione di dati ottenuta nello spazio di dimensionalità ridotta; infatti, con valori particolarmente elevati del parametro, i dati potrebbero risultare equidistanti dai vicini (in strutture di forma tonda), mentre, con valori molto bassi del parametro, i dati potrebbero essere concentrati in strutture molto dense con presenza di pochi outliers molto distanti dal resto della distribuzione; è possibile definire un valore per il parametro, mentre se posto come il default ‘auto’ il learning rate corrisponde al rapporto tra il numero di campioni nei dati di input e l’*early_exaggeration*.
- **n_iter (default: 1000)**: si tratta del numero massimo di iterazioni eseguite dall'algoritmo, e solitamente deve avere un valore superiore a 250 per ottenere risultati significativi.
- **n_iter_without_progress (default: 300)**: si tratta del numero massimo di iterazioni eseguite dall'algoritmo senza ottenere ulteriori progressi, ovvero una sorta di valore per arrivare alla convergenza (e dunque al termine) della procedura; assume valori multipli di 50 (in quanto l'algoritmo verifica solo ogni 50 iterazioni i progressi effettuati).
- **metric (default: ‘euclidean’)**: questo parametro definisce la metrica utilizzata per calcolare la distanza; di default, viene definita come ‘euclidean’ (uguale a ‘l2’), mentre altre opzioni utilizzabili sono ‘manhattan’ (o ‘l1’), ‘cosine’ o ‘precomputed’ (solo se viene data una matrice di distanze come metodo di input dell'algoritmo).
- **n_jobs (default: ‘None’)**: questo parametro indica il numero di operazioni parallele da eseguire per ogni ricerca di vicini (se *n_jobs* viene posto uguale a ‘None’, pari al valore default, è pari a 1, mentre col valore -1 vengono utilizzati tutti i processori);

Con una quantità di parametri non indifferente, la fase di tuning non è semplice e richiede sforzi significativi (uno dei motivi per cui questa tecnica viene utilizzata meno di altri metodi, come la PCA e l'UMAP). Visualizzare i risultati della t-SNE per diverse configurazioni di parametri è fondamentale per verificare quanto bene l'algoritmo cattura la struttura intrinseca dei dati. In particolare, i due parametri che maggiormente contribuiscono nel caratterizzare i risultati della metodologia t-SNE sono la perplessità e il tasso di apprendimento. È fondamentale trovare un equilibrio tra di essi che si adatti alle caratteristiche dei dati e permetta di ottenere risultati significativi.

3.3.3 Passaggi algoritmici

La t-SNE segue una serie di passaggi [Cai 22] [distill.pub] per trasformare dati ad alta dimensionalità in una rappresentazione a dimensionalità inferiore:

-
1. **Normalizzazione/Standardizzazione:** prima di procedere con l'algoritmo, è fondamentale effettuare una trasformazione dei dati per lavorare su dati privi di outliers, e senza features che possano influenzare significativamente i risultati ottenuti per unità di misura e scale differenti dal resto delle features dei dati.
 2. **Calcolo similarità tra coppie:** per ogni coppia di punti dati ad alta dimensionalità, viene calcolata la probabilità condizionata misura della loro similarità; queste probabilità sono basate su un kernel gaussiano, con una distribuzione centrata su ogni punto; si ottiene così una distribuzione di probabilità congiunta su tutte le coppie di dati.
 3. **Calcolo similarità a bassa dimensionalità:** analogamente al passaggio precedente, vengono calcolate le probabilità condizionate per coppie di punti nello spazio a bassa dimensionalità, basate ora sulla distribuzione t di Student, cercando di preservare le similarità trovate al passaggio precedente.
 4. **Ottimizzazione e calcolo del gradiente:** l'algoritmo cerca di minimizzare la divergenza di Kullback-Leibler tra le distribuzioni di probabilità ad alta e bassa dimensionalità, attraverso l'ottimizzazione con la discesa del gradiente (chiamato solitamente gradient descent, ovvero un algoritmo per l'individuazione di un valore di minimo in una funzione di costo); il gradiente della divergenza di Kullback-Leibler viene poi calcolato rispetto ai punti dati a bassa dimensionalità, in modo da ottenere una rappresentazione a bassa dimensionalità ottimale.
 5. **Aggiornamenti del gradient descent:** viene aggiornata iterativamente la rappresentazione a bassa dimensionalità, in modo da minimizzare la divergenza di Kullback-Leibler; il tasso di apprendimento influenza fortemente questa fase del processo iterativo di ottimizzazione.
 6. **Controllo della convergenza:** il processo viene ripetuto fino al raggiungimento della convergenza o, in alternativa, al raggiungimento di un numero predefinito di iterazioni.

3.3.4 Vantaggi e svantaggi

L'algoritmo t-SNE presenta una serie di vantaggi [\[Maaten 08\]](#), tra cui:

- **Efficace nel catturare relazioni non lineari:** la t-SNE eccelle nel catturare strutture non lineari nei dati ad alta dimensionalità, rendendola efficace, soprattutto rispetto ad altre tecniche come la PCA (particolarmente indicate per individuare strutture lineari dei dati), per la visualizzazione di relazioni complesse.
- **Preservazione delle strutture locali:** l'algoritmo è particolarmente adatto alla conservazione di strutture locali e clusters, rendendolo ottimale per la rivelazione di patterns dettagliati nei dati.
- **Robusto rispetto alle variazioni globali:** la t-SNE è relativamente robusta rispetto alle variazioni di scala globali; questo significa che può adattarsi abbastanza bene a dataset con densità o scale molto variabili, anche se operazioni preliminari di normalizzazione o standardizzazione sono comunque consigliate.
- **Visualizzazione di dati ad alta dimensionalità:** l'algoritmo t-SNE fornisce uno strumento potente per la visualizzazione di dati ad alta dimensionalità in due o tre dimensioni, facilitando la fase di analisi esplorativa dei dati.

Tuttavia, l'algoritmo presenta comunque una serie di criticità rilevanti, tra cui:

- **Sensibilità agli iperparametri:** le prestazioni della t-SNE sono sensibili alla scelta degli iperparametri, in particolare la perplessità; valori di perplessità differenti possono portare a visualizzazioni significativamente diverse, dunque la scelta del parametro diventa cruciale per ottenere rappresentazioni chiare e significative.
- **Computazionalmente intensiva:** la t-SNE può essere computazionalmente intensiva, specialmente per dataset di grandi dimensioni; la complessità temporale dell'algoritmo è cubica rispetto al numero di punti dati, rendendolo meno adatto per dataset con un numero elevato di campioni; infatti spesso la

scelta è di utilizzare la t-SNE su campioni del dataset originale, col fine di esplorare le strutture dei dati e verificare i valori ottimali per gli iperparametri.

- **Sensibilità all'inizializzazione casuale:** i risultati della t-SNE possono variare in base alla inizializzazione casuale ad ogni esecuzione; eseguire la t-SNE più volte con diverse inizializzazioni casuali è fondamentale per verificare la coerenza dei risultati, in quanto patterns consistenti tra le esecuzioni aumentano la fiducia nell'affidabilità della visualizzazione.
- **Perdita di struttura globale:** sebbene la t-SNE sia eccellente nel preservare le strutture locali, potrebbe avere difficoltà a mantenere le strutture globali, ed alcune relazioni globali potrebbero essere distorte nella rappresentazione finale a bassa dimensionalità.

3.4 Uniform Manifold Approximation and Projection (UMAP)

3.4.1 Introduzione

Uniform Manifold Approximation and Projection (UMAP) è una tecnica di riduzione della dimensionalità che ha guadagnato rapidamente importanza in vari settori scientifici, tra cui l'apprendimento automatico, l'analisi dei dati e la bioinformatica. Sviluppata da Leland McInnes e John Healy nel 2018, UMAP è emerso rapidamente come una potente alternativa a metodi tradizionali [\[McInnes 20\]](#) come la t-SNE e la PCA.

UMAP è progettata per catturare relazioni complesse all'interno di dati ad alta dimensionalità e proiettarli in uno spazio a dimensioni inferiori preservando le strutture locali e globali. A differenza di metodi lineari come la PCA, UMAP eccelle nel gestire strutture non lineari, rendendolo particolarmente efficace per la visualizzazione di modelli e strutture intricate presenti all'interno dei dati. L'algoritmo sfrutta una combinazione di principi topologici e geometrici per creare una rappresentazione a bassa dimensionalità che conserva la struttura intrinseca dei dati originali.

Il principale scopo di UMAP è la comprensione della geometria sottostante i dati e delle relazioni in spazi ad alta dimensionalità. La tecnica punta ad affrontare alcune limitazioni di altre tecniche di riduzione della dimensionalità, fornendo una soluzione più versatile e interpretabile per l'analisi esplorativa dei dati, la visualizzazione e compiti di apprendimento automatico successivi, come clustering e metodi di regressione o classificazione: infatti la rappresentazione a bassa dimensionalità generata da UMAP dovrebbe migliorare le prestazioni di queste attività, catturando aspetti fondamentali e strutture sottostanti i dati di input.

UMAP ha trovato applicazioni [\[Diaz-Papkovich 20\]](#) [\[Becht 18\]](#) in vari campi scientifici grazie alla sua capacità di gestire strutture complesse e fornire visualizzazioni istruttive. Ecco cinque esempi:

- **Analisi dei dati di sequenziamento RNA a singola cellula in biologia:** UMAP viene utilizzato come metodo di esplorazione dei modelli di espressione genica a livello di singola cellula, per visualizzare ed interpretare popolazioni cellulari distinte, rivelare profili di trascrizione e comprendere l'eterogeneità cellulare.
- **Inserimento di features di immagini per Artificial Vision:** il metodo UMAP è utile per la riduzione dei vettori di caratteristiche d'immagine ad alta dimensionalità in uno spazio a bassa dimensionalità, aiutando a visualizzare e comprendere le relazioni tra le immagini in base alle loro caratteristiche, e facilitando compiti come il clustering delle immagini o il recupero di immagini basato sul contenuto.
- **Filtraggio collaborativo nei sistemi di raccomandazione:** UMAP è impiegato per creare inserimenti a bassa dimensionalità di utenti e oggetti in base alle loro interazioni; il metodo aiuta a catturare le preferenze degli utenti e le somiglianze degli oggetti, migliorando l'efficacia degli algoritmi di filtraggio collaborativo nella generazione di raccomandazioni personalizzate.
- **Chimica molecolare e scoperta di farmaci:** UMAP può essere utilizzato per esplorare e visualizzare le relazioni tra composti chimici, aiutando ad identificare clusters di composti con proprietà simili, guidando le ricerche per la scoperta di nuovi farmaci ed evidenziando composti con strutture molecolari simili.
- **Analisi delle reti sociali:** algoritmi come l'UMAP vengono impiegati per ridurre la dimensionalità dei dati delle reti sociali, preservando la struttura sottostante; in questo modo è possibile visualizzare comunità, identificare nodi influenti e comprendere i modelli di interazione all'interno della rete sociale di interesse.

3.4.2 Iperparametri e tuning

UMAP coinvolge alcuni parametri cruciali [\[McInnes 20\]](#) e regolarli in modo appropriato è essenziale per ottenere risultati significativi:

- **n_neighbors (default: 15)**: questo parametro permette di controllare il bilanciamento tra struttura locale e globale dei dati, limitando la dimensione del vicinato locale utilizzato dall'algoritmo; assume valori compresi solitamente tra 2 (visione molto locale della varietà, tende a formare piccoli clusters di dati simili) e 200 (visione globale della varietà, con composizione globale ben rappresentata a scapito di alcune informazioni locali).
- **n_components (default: 2)**: come per altri metodi di riduzione della dimensionalità, questo parametro indica il numero di componenti principali, quindi anche le dimensioni dello spazio ridotto ottenuto dalla procedura (valori pari a 2 o 3 permettono di osservare graficamente i risultati ottenuti).
- **metric (default: ‘euclidean’)**: questo parametro definisce la metrica utilizzata per calcolare la distanza; può essere definita come una delle distanze di Minkowski (‘euclidean’, ‘manhattan’, ‘chebyshev’...), la distanza ‘cosine’, una distanza normalizzata (ad esempio la distanza ‘minkowski’), una metrica basata su dati binari (come ‘jaccard’ o ‘dice’), o una distanza calcolata; a seconda della natura dei dati, alcune metriche possono preservare meglio le relazioni tra i punti.
- **min_dist (default: 0.1)**: questo parametro indica quanto l'UMAP tende a mappare punti vicini insieme, fornendo una sorta di soglia di distanza minima sotto la quale i punti possono trovarsi nella rappresentazione a dimensioni ridotte; a valori elevati corrispondono dati particolarmente dispersi, mentre col calare del valore diminuisce anche la distanza tra punti nello spazio ridotto.

È dunque fondamentale visualizzare i risultati di UMAP per diverse configurazioni dei parametri e verificare quanto bene l'algoritmo catturi la struttura intrinseca dei dati, motivo per cui strumenti di visualizzazione come grafici a dispersione o heatmaps possono fornire insights utili per l'impatto delle scelte dei parametri.

3.4.3 Passaggi algoritmici

UMAP segue una serie di fasi [\[McInnes 20\]](#) per trasformare i dati ad alta dimensionalità in una rappresentazione a bassa dimensionalità:

1. **Pulizia e standardizzazione/normalizzazione**: prima di procedere con i passaggi veri e propri dell'algoritmo, è fondamentale la fase di pulizia dei dati e la trasformazione per ridurre sia l'effetto degli outliers, sia le differenze date da unità di misura e scale dei dati differenti, che possono inficiare i risultati finali dell'algoritmo.
2. **Costruzione dei vicini ad alta dimensionalità**: per ogni punto dati in input, vengono identificati i suoi vicini (numero controllato dal parametro `n_neighbors`) nella rappresentazione ad alta dimensionalità, in base alla metrica di distanza scelta.
3. **Apprendimento e costruzione della rappresentazione topologica fuzzy**: UMAP costruisce una rappresentazione topologica fuzzy dei dati, enfatizzando sia strutture locali che globali, in modo da minimizzare una funzione di perdita che quantifichi la discrepanza tra la rappresentazione topologica fuzzy nello spazio ad alta dimensionalità e la rappresentazione a bassa dimensionalità (questa funzione considera sia relazioni locali che globali tra i punti dati); UMAP inoltre incorpora un parametro che impedisce ai punti dati di essere troppo vicini (con valore dato da ‘min_dist’) nello spazio a bassa dimensionalità.
4. **Ottimizzazione rappresentazione a bassa dimensionalità**: l'algoritmo UMAP termina con l'ottimizzazione della rappresentazione topologica fuzzy ottenuta al passaggio precedente, tramite l'aggiornamento iterativo della rappresentazione a bassa dimensionalità per minimizzare la funzione di perdita, calcolando il descent gradient; questo procedimento permette di catturare l'incertezza e la “fuzziness” nelle relazioni ad alta dimensionalità, contribuendo alla capacità di UMAP di gestire strutture complesse.

3.4.4 Vantaggi e svantaggi

UMAP è quindi un algoritmo di riduzione di dimensionalità con una serie di aspetti positivi [McInnes 20], tra cui:

- **Preservazione di strutture locali e globali:** UMAP eccelle nella preservazione di strutture locali e globali in dati ad alta dimensionalità, fornendo una rappresentazione completa delle relazioni intrinseche ai dati in input.
- **Gestione di strutture non lineari:** l'algoritmo UMAP è particolarmente adatto alla cattura di relazioni non lineari, superando le limitazioni presenti in tecniche lineari come la PCA.
- **Efficienza per grandi insiemi di dati:** UMAP è spesso più efficiente dal punto di vista computazionale per grandi dataset rispetto ad altri metodi di riduzione della dimensionalità come t-SNE, rendendolo particolarmente adatto a gestire quantità considerevoli di dati con numero elevato di features.
- **Semplicità nell'interpretabilità:** le rappresentazioni a bassa dimensionalità ottenute dall'algoritmo sono spesso più interpretabili, consentendo una migliore comprensione delle strutture sottostanti nei dati.

Tuttavia, l'algoritmo UMAP presenta comunque una serie di svantaggi, tra cui:

- **Sensibilità dei parametri:** le prestazioni di UMAP sono fortemente condizionate dalla scelta dei parametri, specialmente il numero di vicini (`n_neighbors`) e la distanza minima (`min_dist`); inoltre, trovare valori ottimali per i parametri può richiedere tempo, molta esperienza nel campo e parecchia capacità computazionale.
- **Impatto della Random Initialization:** i risultati ottenuti con UMAP possono variare fortemente in base alla Random Initialization, quindi eseguire l'algoritmo più volte con differenti seeds può produrre proiezioni diverse; in questo modo, per verificare la validità del risultato ottenuto è necessario ripetere la procedura più volte con particolare attenzione al tuning dei parametri.
- **Non semplice interpretazione dei risultati:** come per altre tecniche di riduzione della dimensionalità, interpretare il significato esatto di distanze e clusters nella visualizzazione di UMAP può essere complesso, e bisogna fare attenzione a non sovrastimare i risultati; UMAP manca inoltre di un'interpretazione probabilistica chiara, rendendo difficile fornire garanzie teoriche sulla conservazione di determinate relazioni durante la riduzione della dimensionalità.

3.5 Altri metodi di riduzione della dimensionalità

Nel progetto in studio, è stata valutata la possibilità di utilizzare le seguenti tecniche [\[Maaten 09\]](#) tra i metodi di riduzione della dimensionalità.

3.5.1 Kernel Principal Component Analysis (kernel PCA)

La Kernel PCA [\[Lee 07\]](#) [\[Ghodsi 06\]](#) è una tecnica di riduzione di dimensionalità che deriva dalla PCA utilizzando dei kernel per riduzione di dimensionalità non lineare. Infatti, la PCA classica è efficace per identificare le direzioni principali di massima varianza nei dati, ma può avere limitazioni quando i dati sono distribuiti in modo non lineare nello spazio dimensionale: la kernel PCA permette di risolvere questo problema, utilizzando una funzione kernel per proiettare i dati di input in uno spazio di dimensioni più grandi di quello originale.

In breve, la Kernel PCA consente di affrontare la non linearità nei dati, aprendo la possibilità di catturare relazioni più complesse e identificare strutture nascoste. Alcuni esempi comuni di funzioni kernel includono il kernel lineare, il kernel polinomiale, il kernel gaussiano (o RBF - Radial Basis Function) o il kernel coseno...

3.5.2 Independent Component Analysis (ICA)

L'Analisi delle Componenti Indipendenti (ICA, Independent Component Analysis) è una tecnica computazionale utilizzata per separare un segnale multivariato in componenti additive e indipendenti. A differenza dell'Analisi delle Componenti Principali (PCA), che si concentra sulla decorrelazione dei dati, l'ICA mira ad individuare fonti statisticamente indipendenti che contribuiscono ai dati osservati. L'ICA è impiegata in vari campi, tra cui l'elaborazione dei segnali, l'analisi delle immagini, la separazione cieca delle sorgenti e l'estrazione delle caratteristiche.

L'ICA è principalmente utilizzata per separare una serie di segnali sorgente nelle rispettive componenti originali e indipendenti. Il metodo di riduzione della dimensionalità si basa sull'assunzione che le sorgenti dati in input siano statisticamente indipendenti e non gaussiane: ciò la rende particolarmente adatta a situazioni in cui i dati sono intrinsecamente indipendenti ma non necessariamente distribuiti in modo normale. Questa tecnica può rivelare modelli o componenti nascosti nei dati, migliorando potenzialmente le prestazioni degli algoritmi successivi.

3.5.3 Singular Value Decomposition (SVD)

La SVD (Singular Value Decomposition, o Decomposizione a Valori Singolari) è una tecnica di algebra lineare utilizzata per la fattorizzazione di matrici. Essa decomponete una data matrice in tre matrici separate, fornendo una comprensione della struttura e delle proprietà dei dati originali. La SVD è ampiamente impiegata in diversi campi, tra cui l'analisi dei dati, l'elaborazione di immagini, l'elaborazione di segnali, i sistemi di raccomandazione e la riduzione della dimensionalità.

La SVD scomponete una matrice A in tre matrici separate U, Σ (Sigma) e V^T (la trasposta di V), dove U e V sono matrici ortogonali e Σ è una matrice diagonale. Questa fattorizzazione consente di esprimere la matrice originale come una combinazione lineare dei suoi componenti, semplificando l'analisi e la manipolazione dei dati. In questo modo, la SVD può essere utilizzata per ridurre la dimensionalità dei dati, potendo dunque rappresentare i dati in uno spazio a dimensione inferiore.

3.5.4 Non-Negative Matrix Factorization (NMF)

La Decomposizione di Matrici Non-Negative (NMF, Non-Negative Matrix Factorization) è una tecnica di riduzione della dimensionalità ed estrazione delle caratteristiche utilizzata per scomporre una matrice non-negativa nel prodotto di due matrici non-negative di dimensione inferiore. L'NMF viene utilizzata in svariati ambiti per la sua capacità di estrarre modelli significativi e interpretabili dai dati, soprattutto quando i dati sono non negativi.

L'NMF impone la non-negatività su entrambe le matrici dei fattori, rendendola adatta a set di dati in cui tutti i valori sono intrinsecamente non negativi, come immagini, testi e dati di espressione genica. Questo vincolo porta a risultati di scomposizione più intuitivi e interpretabili. L'NMF riduce la dimensionalità dei dati approssimando la matrice originale con due matrici di dimensione inferiore. Ciò può essere vantaggioso per la compressione dei dati, la visualizzazione e l'accelerazione delle analisi successive. In contesti come la modellazione degli argomenti, le colonne delle matrici dei fattori possono essere interpretate come argomenti e i pesi come l'importanza degli argomenti nei documenti.

3.5.5 Criticità dei metodi kernel PCA, ICA, SVD e NMF

I metodi appena descritti, tuttavia, non sono stati utilizzati nel progetto per i seguenti motivi:

- Le operazioni tipiche della PCA non possono essere riprodotte con la Kernel PCA; infatti, la creazione di uno spazio dimensionale di dimensioni superiori a quello dei dati in input e i passaggi successivi dell'algoritmo necessitavano di dispositivi con capacità computazionale di gran lunga superiore rispetto a quelli a disposizione (il metodo creava una matrice di dimensione pari a 608 GB), e l'impossibilità di utilizzare strumenti di lavoro in cloud per ovviare a questi problemi, data la necessità di elevati livelli di privacy per i dati molto sensibili a disposizione, hanno portato alla decisione di concentrarsi su altri metodi.
- Anche per quanto riguarda gli altri metodi, ovvero ICA, SVD e NMF, la natura dei dati di input (come grandezza del campione e numero elevato di features) non ha permesso il passaggio fondamentale, presente in tutti questi algoritmi, della decomposizione della matrice di dati in input: infatti, le matrici formate avrebbero avuto dimensioni di circa 300 GB. Per questo motivo, sarebbe stato necessario o un campionamento molto importante (con conseguente perdita di gran parte dei dati in input) o l'utilizzo di strumenti in cloud (impossibile per le necessità di privacy legate alla natura dei dati in studio).

3.6 Risultati e discussione riduzione dimensionalità

Sono state applicate diverse tecniche di riduzione di dimensionalità tra quelle descritte nella sezione precedente.

In particolare, prima dell'esecuzione di questi metodi, è stata effettuata una nuova fase di pulizia dei dati, con una serie di passaggi atti a rendere i dati adatti all'applicazione delle tecniche:

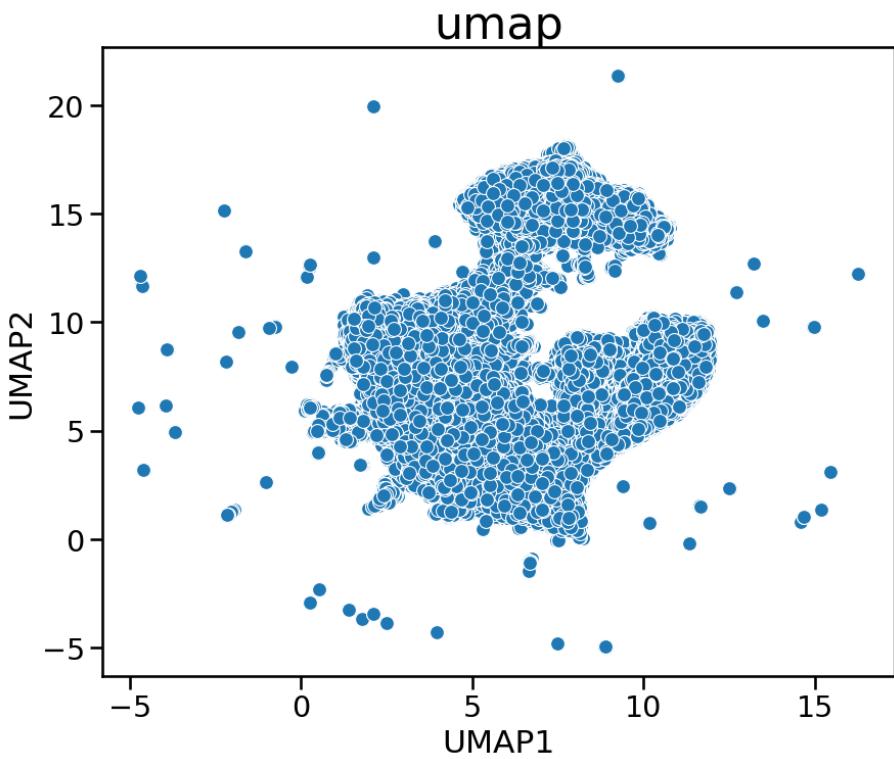
- Avendo nel dataset anche campioni BIS e di controllo, è stato preso un solo record per ogni neonato (identificato univocamente con la variabile *id*) che coincide con l'ultimo record dal punto di vista del campione ricevuto nel dataset, indicato dalla variabile *SamTimeReceived*.
- Sono state escluse le variabili relative a date, in quanto non sono state considerate importanti per i fini della riduzione di dimensionalità.
- Sono state escluse, tra le variabili quantitative, le colonne con percentuali di valori mancanti sopra il 5%, portando il totale a 44 features.
- Allo stesso modo, sono state rimosse le variabili qualitative con percentuali di valori mancanti sopra al 5%, in modo da mantenere nel dataset 22 colonne relative ad informazioni qualitative; inoltre, sono state rimosse le variabili *id*, *Hospital*, *City*, *Etnicity* e *SampleBarcodeAnonymized*, in quanto non sono state considerate importanti per i fini della riduzione di dimensionalità, portando il totale di features qualitative a 17.
- I dati quantitativi sono stati scalati con uno *standard scaler*, per permettere il confronto tra dati con unità di misura e scale differenti e limitare gli effetti determinati dalla presenza di outliers, che rischiavano di inficiare fortemente i risultati.
- Sono state create, tramite il metodo `.get_dummies` del pacchetto *Pandas*, le variabili dummies relative alla variabili qualitative conservate nel dataset fino a questo momento: il numero di colonne qualitative dunque sale a 25.
- Sono state a questo punto escluse tutte le osservazioni con presenza di dati mancanti: infatti, su richiesta dei clinici esperti di dominio, è stato scelto di evitare metodi quali l'imputazione di valori mancanti, per non inficiare informazioni delicate quali i dati clinici dei neonati, a maggior ragione avendo comunque a disposizione un dataset di dimensioni comunque molto significative.

Il dataset finale ottenuto è dunque formato da 272114 osservazioni, con 25 features qualitative e 44 variabili quantitative per un totale di 69. Su questo dataset verranno applicati non solo i metodi di riduzione di dimensionalità, ma anche le tecniche di cluster analysis dei capitoli successivi.

Sono stati riportati i risultati ottenuti con una serie di valori scelti dei parametri dei differenti metodi, ma per questi risultati sono state necessarie diverse verifiche sui valori più rilevanti (alcuni risultati non significativi non sono infatti stati riportati). La scelta di visualizzazioni bidimensionali è giustificata dalla necessità di risultati chiari e facilmente leggibili, a prescindere dalla perdita superiore di informazione dovuta dalla dimensionalità minore in output dei metodi applicati.

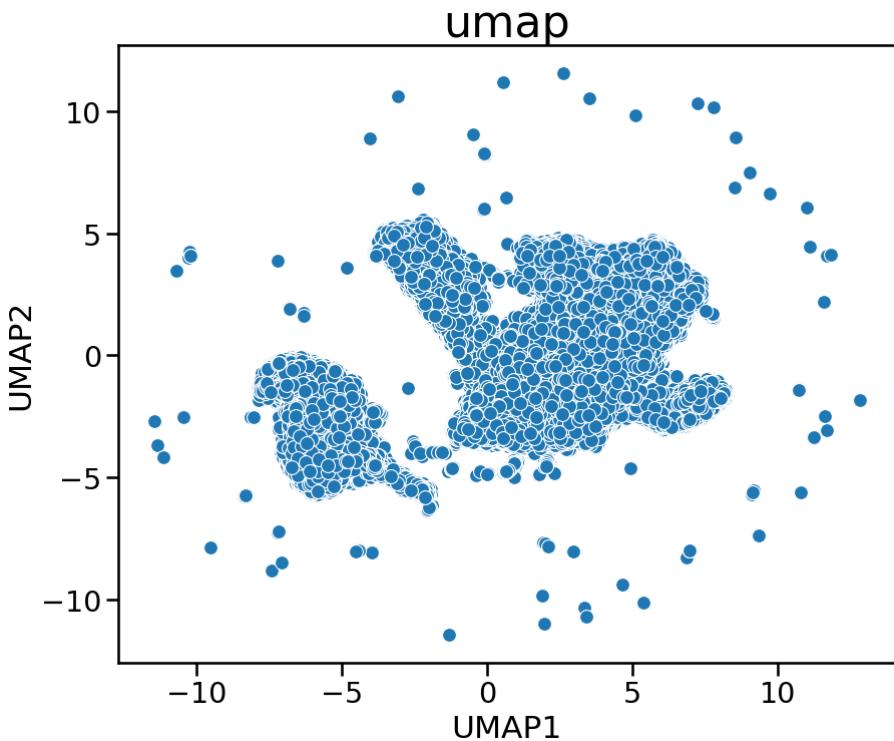
3.6.1 UMAP

È stata applicata la metodologia UMAP [\[umap-learn.readthedocs.io\]](https://umap-learn.readthedocs.io), modificando i parametri quali *n_neighbors* (controlla quanto UMAP mantiene della struttura locale rispetto alla struttura globale dei dati, con valori bassi per forte struttura locale), *min_dist* (controlla quanto distano i punti in rappresentazioni a basse dimensioni, con valori bassi che indicano clusters molto più fitti), *n_components* (per definire la dimensionalità dello spazio risultante dall'algoritmo: nel nostro caso, sempre scelto pari a 2) e *metric* (come viene calcolata la distanza tra punti: nel nostro caso sempre mantenuta pari a euclidea), abbiamo ottenuto i seguenti risultati.



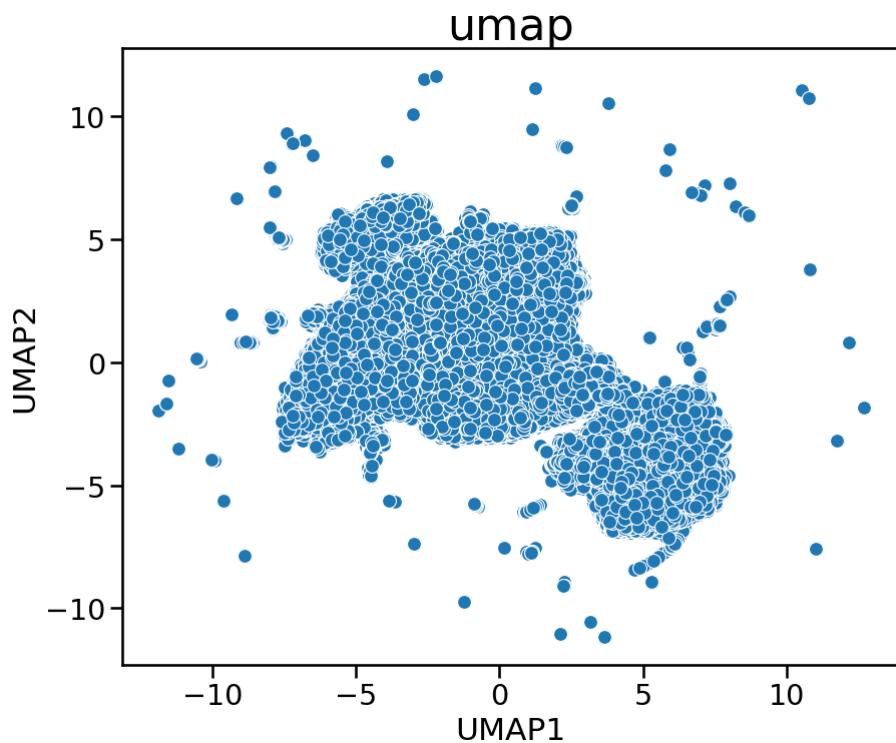
```
umap_reducer = umap.UMAP(n_neighbors=5, min_dist=0.05, n_components=2,  
metric='euclidean')
```

Figure 3.9: scatterplot UMAP(*n_neighbors*=5, *min_dist*=0.05)



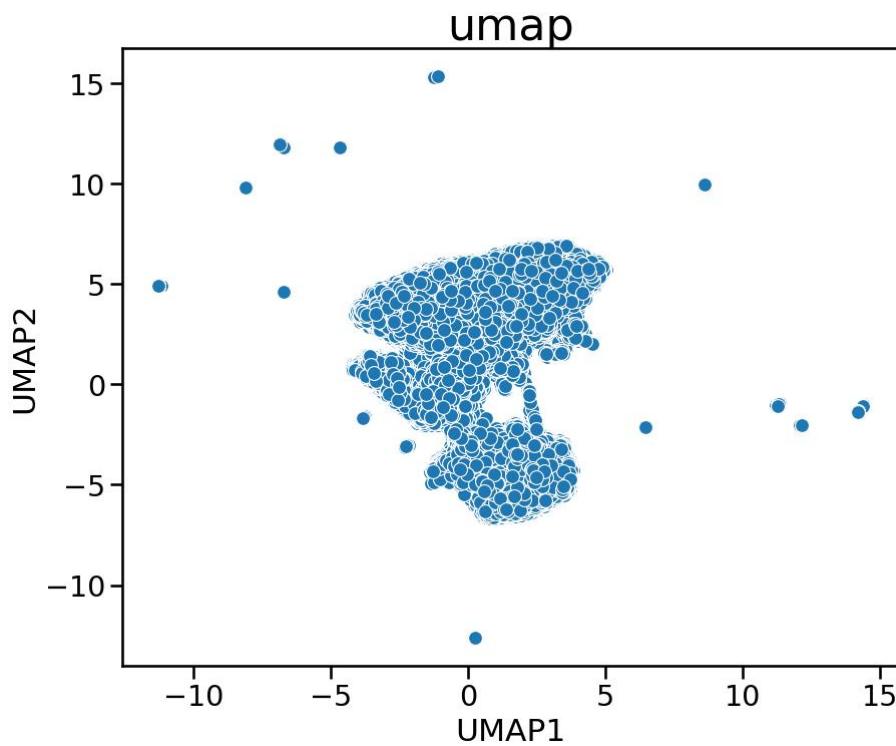
```
umap_reducer = umap.UMAP(n_neighbors=5, min_dist=0.01, n_components=2,  
metric='euclidean')
```

Figure 3.10: scatterplot UMAP(*n_neighbors*=5, *min_dist*=0.01)



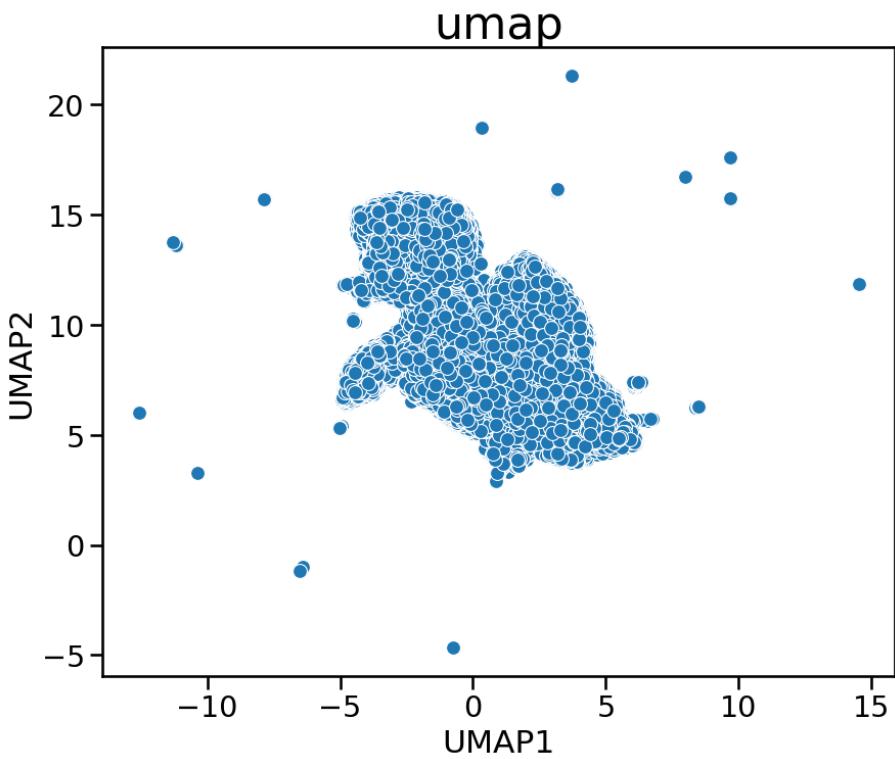
```
umap_reducer = umap.UMAP(n_neighbors=5, min_dist=0.1, n_components=2,
metric='euclidean')
```

Figure 3.11: scatterplot UMAP($n_neighbors=5$, $min_dist=0.1$)



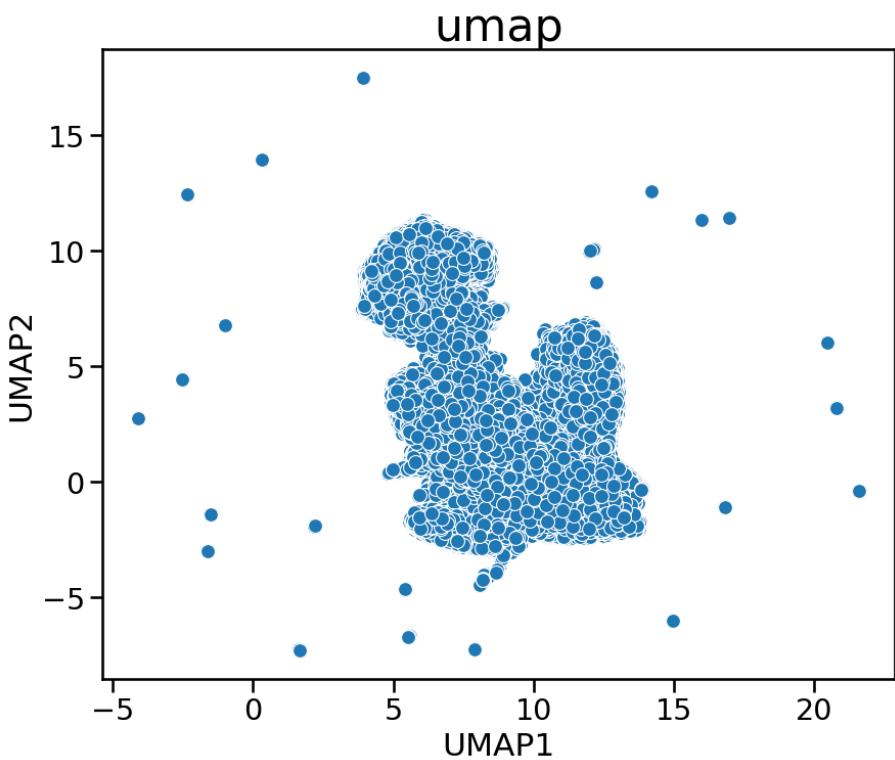
```
umap_reducer = umap.UMAP(n_neighbors=50, min_dist=0.1, n_components=2,
metric='euclidean')
```

Figure 12: scatterplot UMAP($n_neighbors=50$, $min_dist=0.1$)



```
umap_reducer = umap.UMAP(n_neighbors=30, min_dist=0.1, n_components=2,  
metric='euclidean')
```

Figure 3.13: scatterplot UMAP($n_{neighbors}=30$, $min_dist=0.1$)



```
umap_reducer = umap.UMAP(n_neighbors=15, min_dist=0.1, n_components=2,  
metric='euclidean')
```

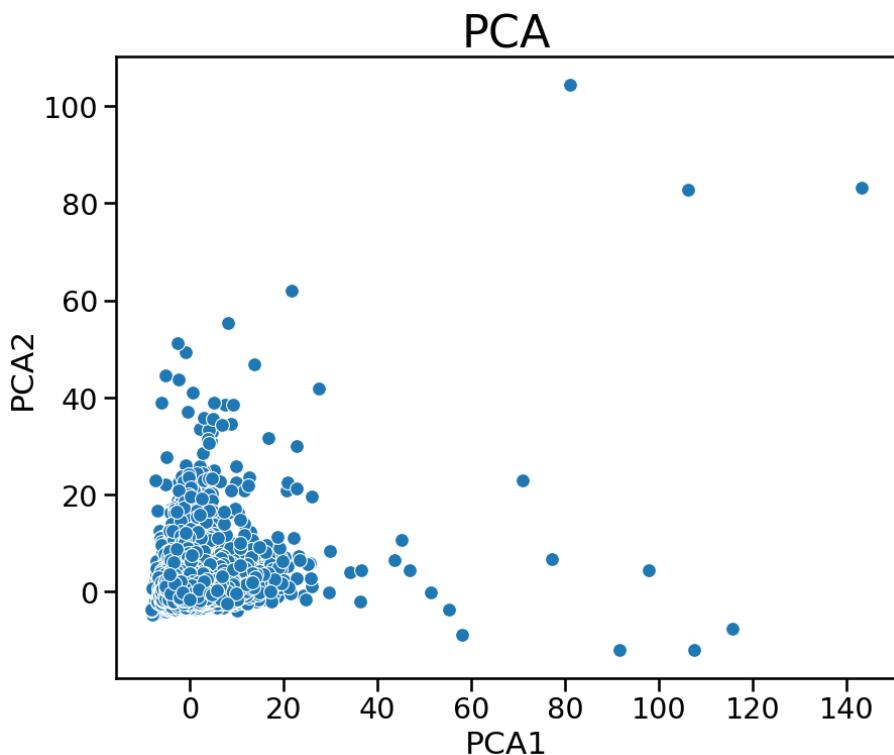
Figure 3.14: scatterplot UMAP($n_{neighbors}=15$, $min_dist=0.1$)

I grafici ottenuti col metodo UMAP variano discretamente per valori differenti dei parametri `n_neighbors` e `min_dist`: all'aumentare del primo, si ottengono gruppi di osservazioni più compatti e con forte struttura locale; per il secondo parametro, a valori più bassi corrispondono clusters più fitti, mentre al crescere tendono a crearsi gruppi di istanze meno densi internamente.

Le differenze di risultati con le implementazioni eseguite possono essere utili all'individuazione di gruppi di osservazioni con caratteristiche simili, in base alla struttura dei dati analizzati.

3.6.2 PCA

Dopo l'algoritmo UMAP, è stata applicata la metodologia PCA [\[scikit-learn.org\]](#). Anche provando a modificare alcuni parametri (come `whiten`, `svd_solver`, `tol`, `iterated_power`, `n_oversamples` o `power_iteration_normalizer`) il risultato della riduzione di dimensionalità non cambia, dunque viene riportato il solo risultato ottenuto con questa tecnica.

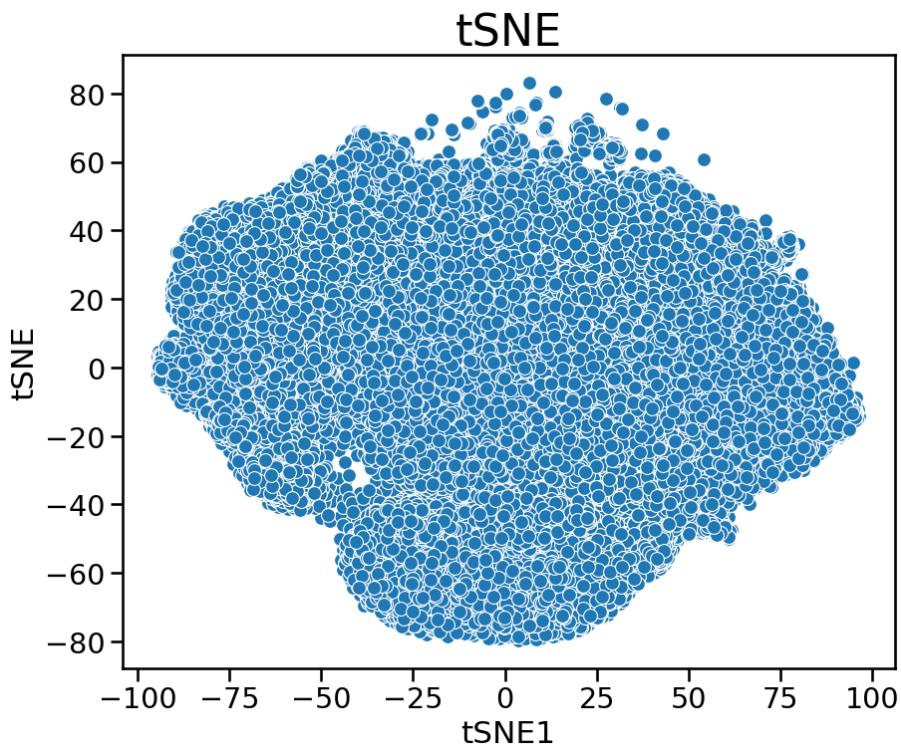


```
pca = PCA(n_components=2)
Figure 15: scatterplot PCA(n_components=2)
```

Dal punto di vista grafico, la visualizzazione non sembra ottimale per catturare clusters differenti, in quanto la PCA sembra restituire un grosso cluster molto denso e compatto, con alcune osservazioni che si allontanano dalla nuvola generale dei dati: metodi di cluster analysis che permettono di individuare clusters densi e ben separati gli uni dagli altri non sono probabilmente adatti ad una visualizzazione del genere...

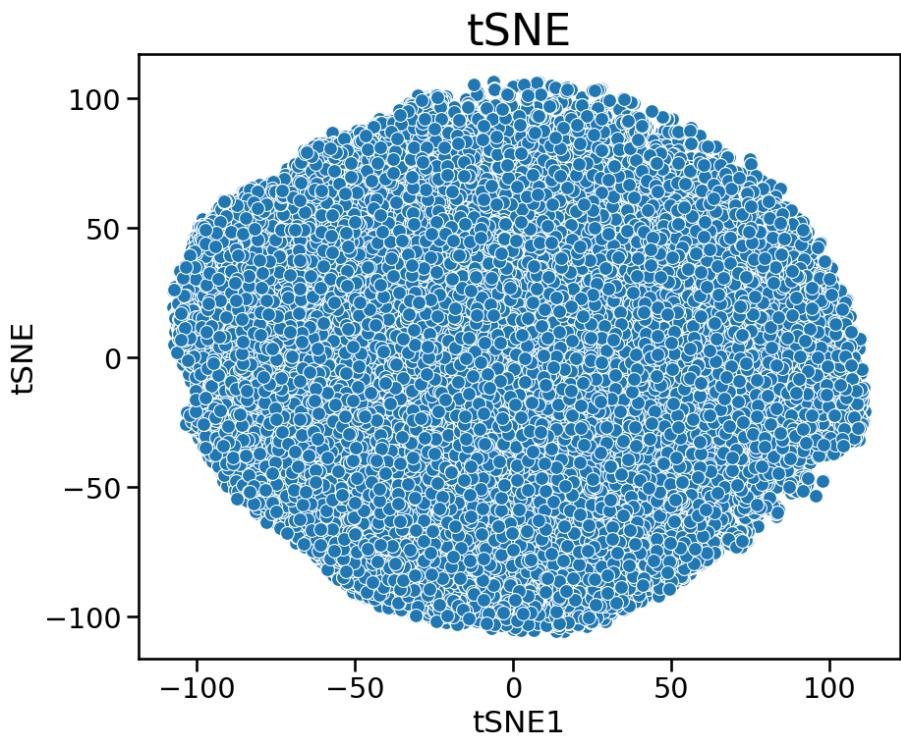
3.6.3 t-SNE

Infine, l'ultimo metodo applicato è stata la t-SNE [\[scikit-learn.org\]](#), in cui sono stati modificati parametri quali la perplexity (influenza l'equilibrio tra la conservazione delle strutture globali e locali nei dati) ed `early_exaggeration` (parametro che definisce quanto i campioni nei clusters vengono riprodotti vicini nello spazio di dimensionalità ridotto).



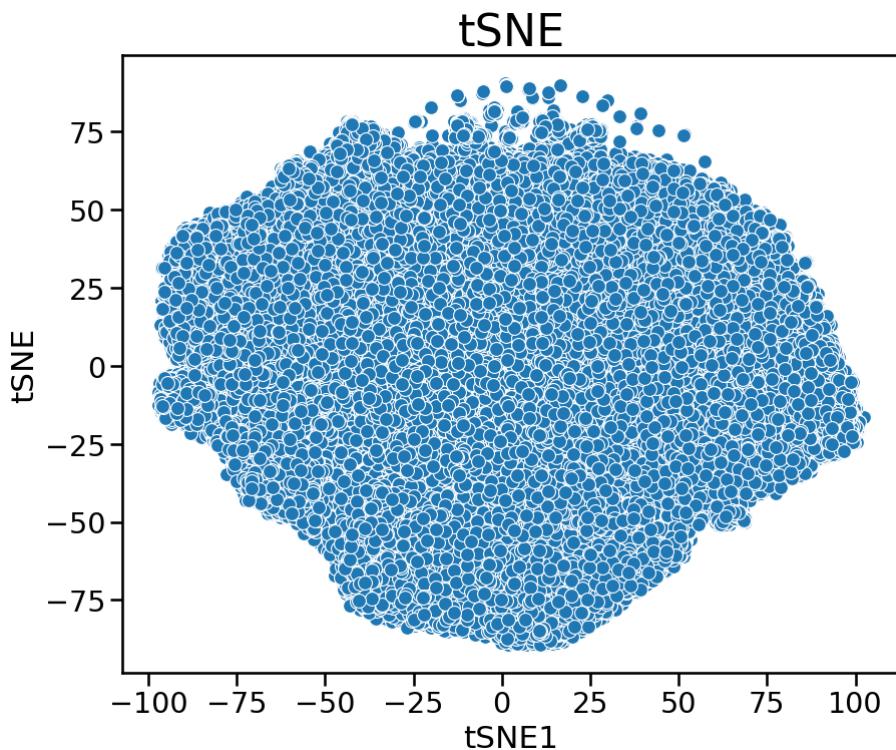
```
tsne = TSNE(n_components=2, perplexity=5, early_exaggeration=12)
```

Figure 3.16: scatterplot tSNE(perplexity = 5, early_exaggeration = 12)



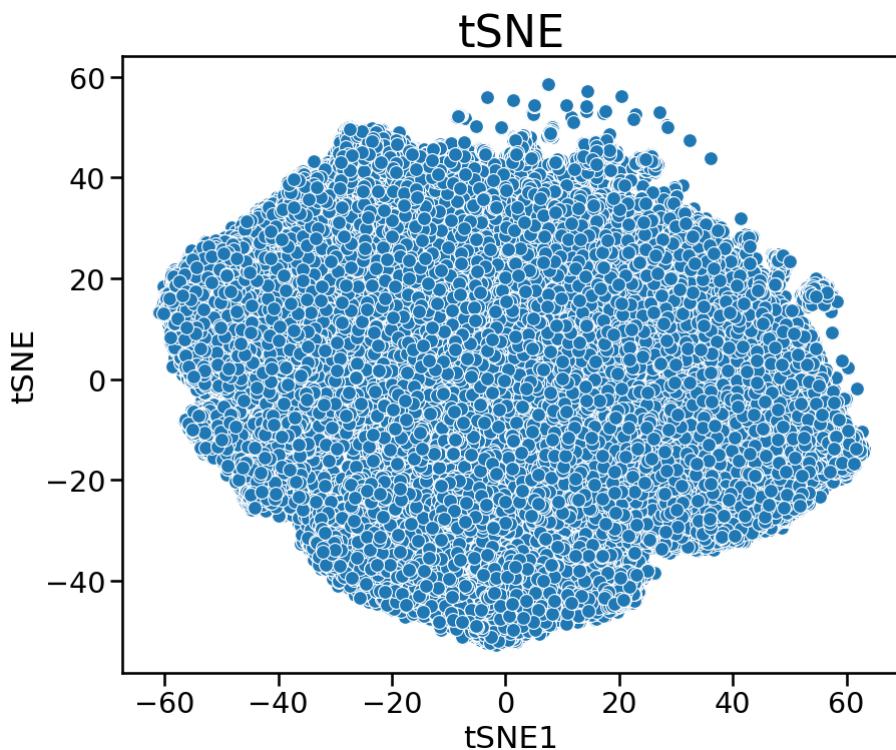
```
tsne = TSNE(n_components=2, perplexity=50, early_exaggeration=12)
```

Figure 3.17: scatterplot tSNE(perplexity = 50, early_exaggeration = 12)



```
tsne = TSNE(n_components=2, perplexity=30, early_exaggeration=12)
```

Figure 3.18: scatterplot tSNE(perplexity = 30, early_exaggeration = 12)



```
tsne = TSNE(n_components=2, perplexity=30, early_exaggeration=50)
```

Figure 3.19: scatterplot tSNE(perplexity = 5, early_exaggeration = 50)

I grafici ottenuti col metodo t-SNE hanno piccole variazioni al variare dei valori dei parametri *perplexity* ed *early_exaggeration*: solo una piccola parte della visualizzazione, a margine della nuvola di dati che comprende tutte le osservazioni, varia con valori leggermente più distanti dagli altri al crescere dell' *early_exaggeration* e con valori bassi di *perplexity*.

In generale, comunque, i clusters ottenuti a partire dai dati ridotti con t-SNE difficilmente vedranno la formazione di gruppi molto densi internamente e ben separati dagli altri, visto il risultato ottenuto con l'algoritmo.

Infine, bisogna tenere conto che si tratta di una tecnica computazionalmente molto dispendiosa, in termini di requisiti dei dispositivi e di tempi di esecuzione, il che rende ulteriormente difficile l'operazione di tuning dei parametri.

4. Cluster Analysis

4.1.1 Introduzione

La cluster analysis [Banks 04] [Vark 04] [Brito 07], tecnica fondamentale nell'analisi dei dati e nel riconoscimento di pattern in dataset di medio/grandi dimensioni, svolge un ruolo cruciale nell'identificare strutture intrinsecamente presenti nei dataset studiati. Raggruppando entità simili in clusters, viene facilitata l'esplorazione di pattern, relazioni e insights in vari ambiti.

4.1.2 Cenni storici

Le origini storiche della cluster analysis [Murtagh 13] possono essere ricondotte ai primi anni del XX secolo. Lo psicologo Charles Spearman, nel 1904, introdusse l'analisi fattoriale, un precursore della cluster analysis, per identificare fattori sottostanti ai dati di test psicologici. Tuttavia, solo negli anni '50 il termine "cluster analysis" acquisì finalmente rilevanza. Lo psicologo Robert L. Thorndike e la matematica Dorothy Mahalanobis contribuirono significativamente al suo sviluppo, applicando la cluster analysis a dati psicologici e biologici.

Negli anni '60, alcuni matematici come Jardine e Sibson ampliarono ulteriormente le tecniche di applicazione della cluster analysis introducendo i primi metodi di clustering gerarchico. Negli anni '70 si svilupparono metodi non gerarchici, tra cui K-means e Partizionamento Attorno ai Mediodi (PAM). Da allora, la cluster analysis è diventata essenziale in svariate discipline, tra cui la statistica, la biologia e l'informatica.

4.1.3 Obiettivi della cluster analysis

La cluster analysis mira a categorizzare un dataset in gruppi, o clusters, basati sulla similarità dei suoi elementi. Gli obiettivi principali includono:

- **Riconoscimento dei pattern:** la cluster analysis viene spesso utilizzata per identificare strutture o pattern intrinseci nei dati, non direttamente osservabili a partire da una prima osservazione dei dati.
- **Compressione dei dati:** la cluster analysis è fondamentale per semplificare dataset complessi, permettendo di raggruppare elementi simili e distinguere gli elementi di gruppi differenti.
- **Rilevamento delle anomalie:** le tecniche di cluster analysis possono essere utilizzate per evidenziare outliers che non seguono i pattern dei clusters, in modo da riconoscere anomalie nel corpus di dati a disposizione.
- **Supporto alle decisioni:** la cluster analysis è fondamentale per aiutare nella presa di decisioni, fornendo indicazioni sulle relazioni tra i dati.

4.1.4 Parametri chiave nella cluster analysis

Sono molte le tecniche di cluster analysis sviluppate negli anni, e si differenziano come ambiti di applicazione (alcune tecniche sono più indicate per certe tipologie di dati), come formato dei dati a disposizione (alcune tecniche lavorano meglio con dimensioni e quantità di dati differenti, e con tipologie di dati a disposizione molto diverse) e come parametri da implementare in base alla tecnica scelta. Tuttavia, alcuni parametri sono comuni ad una moltitudine di tecniche di cluster analysis, tra cui:

- **La metrica di distanza:** la scelta della metrica di distanza influenza significativamente la formazione dei clusters; infatti, metriche comuni come la famiglia di distanze di Minkowski, la distanza euclidea, la distanza di Manhattan e la distanza coseno permettono di riconoscere efficacemente le strutture

sottostanti i dati, evitando che la scelta della distanza influenzi negativamente l'algoritmo con la creazione di clusters poco significativi e con dati poco consistenti al loro interno.

- **Il numero di clusters (k):** In diversi metodi, come il K-means, il BIRCH e lo Spectral Clustering, determinare il numero ottimale di clusters è cruciale; è possibile utilizzare una serie di tecniche, come il metodo del gomito o l'analisi di silhouette, che aiutano a trovare il valore k appropriato al caso in studio.
- **Criteri di convergenza** (per i metodi iterativi): per i metodi iterativi che richiedono una scelta di parametro per raggiungere la convergenza del metodo, come il K-means e lo Spectral Clustering; tipicamente, la procedura termina al raggiungimento del valore prestabilito, e fornisce come risultato finale l'ultimo set di clusters calcolato.

Inoltre, ogni tecnica ha una serie di parametri caratteristici della metodologia stessa, sempre fondamentali per ottenere risultati validi e significativi, ognuno con criteri di scelta ed indicazioni da seguire per l'implementazione dei valori e metodi. La cluster analysis può essere applicata a vari tipi di dati, inclusi dati quantitativi, qualitativi o di tipo misto: la scelta dell'algoritmo di clustering dipende dal tipo di dati. È importante considerare anche che dataset di grandi dimensioni possono richiedere algoritmi di clustering più sofisticati e metodi computazionali efficienti, mentre al contrario dataset più piccoli possono essere particolarmente suscettibili all'overfitting.

Infine, è fondamentale fare attenzione ad alcuni aspetti legati alla natura dei dati utilizzati, che possono condizionare in maniera significativa i risultati ottenuti, oltre ai tempi di esecuzione delle metodologie di cluster analysis:

- **Normalizzazione/standardizzazione:** assicurarsi che le variabili siano su scale simili è fondamentale, infatti questo impedisce alle features con scale più grandi di dominare il processo di clustering, condizionando non solo la creazione di clusters (un punto particolarmente lontano dalla distribuzione potrebbe essere visto come singolo cluster, condizionando l'intero dataset) ma anche la composizione di gruppi già esistenti (gruppi particolarmente distanti potrebbero essere considerati uniti per la presenza di outliers); inoltre, la presenza di outliers con valori particolarmente discostati dalle distribuzioni è fonte di rumore per i risultati dell'algoritmo di clustering.
- **Selezione delle variabili di interesse:** non tutte le caratteristiche dei dati devono essere considerate rilevanti per il clustering, infatti è fondamentale evitare rumore e informazioni non pertinenti che possono solo condizionare i dati; per questo motivo, le fasi di pulizia dei dati, studio delle distribuzioni degli stessi, imputazione di dati mancanti, operazioni per lo studio di collinearità e riduzione della dimensionalità sono cruciali per ottenere risultati validi e significativi.
- **Scelta delle metriche di valutazione:** utilizzare metriche di valutazione adeguate per valutare la validità dei clusters ottenuti, come il punteggio della silhouette o l'indice di Davies-Bouldin, è fondamentale; trattandosi di apprendimento non supervisionato, non è possibile verificare i risultati ottenuti con delle labels pre esistenti, dunque la scelta dell'indice per valutare i risultati ottenuti è cruciale.
- **Gestione dei dati mancanti:** ovviamente, vista la centralità dei dati nel metodo, implementare valide strategie per trattare i valori mancanti, come l'imputazione o direttamente la rimozione, è fondamentale.

4.1.5 Applicazioni della cluster analysis

Le tecniche di cluster analysis sono nate agli inizi del XX secolo, come conseguenza della necessità di trovare gruppi di dati con caratteristiche simili, in assenza di labels specifiche già all'interno dei dati, in

molti contesti differenti. Trattandosi dunque di tecniche e metodologie con una forte componente pratica, è facile immaginare che gli ambiti di ricerca siano cresciuti esponenzialmente col passare del tempo. Tra gli utilizzi più comuni della cluster analysis ora:

- **Segmentazione dei clienti nel marketing:** la cluster analysis è uno strumento che aiuta sempre di più gli esperti di marketing ad identificare segmenti distinti di clienti basati su comportamenti di acquisto, caratteristiche demografiche ed anagrafiche o preferenze. Infatti, in una società sempre più inserita nel contesto dei social media, questi strumenti devono essere utilizzati in maniera mirata per creare valore per le aziende. Dunque la cluster analysis in ambito marketing consente alle aziende di organizzare strategie di marketing mirate e approcci personalizzati, per quanto riguarda suggerimenti di prodotti e servizi, un rapporto cliente-azienda più stretto con molta più informazione, condivisione di opinioni, campagne di mercato e sondaggi mirati a catturare interessi e preferenze della clientela.
- **Segmentazione delle immagini in Artificial Vision:** nell'ambito del trattamento delle immagini, la cluster analysis è impiegata per la segmentazione, ovvero la suddivisione di un'immagine in regioni con caratteristiche visive simili, fase fondamentale per operazioni come il riconoscimento di immagini (creando labels per gruppi di immagini considerati simili in base ai soggetti e ai topics principali), comprensione della scena e la compressione delle immagini, riducendo gruppi di pixel di immagini in singoli gruppi per ridurre le dimensioni dei files limitando al minimo la perdita di informazioni.
- **Classificazione dei documenti nel NLP** (o Natural Language Processing): il Natural Language Processing è un task di intelligenza artificiale che si sta sviluppando sempre di più negli ultimi anni con l'esplosione dello sviluppo di tecniche di intelligenza artificiale; al suo interno racchiude sia tecniche di apprendimento supervisionato (come la text classification) e tecniche non supervisionate (come il topic clustering, utilizzato per raggruppare tipologie di documenti simili, facilitando compiti come la categorizzazione dei documenti, l'analisi del sentiment generale e il recupero delle informazioni in grandi corpus di testi).
- **Rilevamento delle frodi in finanza:** in finanza, le tecniche di clustering aiutano ad individuare comportamenti insoliti e dunque sospetti nelle transazioni finanziarie, consentendo l'identificazione rapida di attività fraudolente. Clusters insoliti possono indicare comportamenti anomali, come spese particolarmente elevate per una tipologia di cliente, indice di un probabile furto di dati o di carta di credito.
- **Analisi delle reti sociali:** in una società dove i social media assumono un ruolo sempre più di rilievo, la cluster analysis viene utilizzata per identificare comunità all'interno delle reti sociali; l'analisi delle reti sociali permette di comprendere le strutture di rete, di seguire il flusso delle informazioni e di riconoscere i nodi influenti all'interno delle comunità.
- **Riconoscimento dei pattern nelle scienze climatiche:** la cluster analysis è stata spesso impiegata per categorizzare regioni con pattern climatici simili, in modo da comprendere la variabilità climatica, cercare di prevedere tendenze e formulare politiche ambientali.
- **Controllo di qualità:** nella produzione di aziende, la cluster analysis ha permesso lo sviluppo di tecniche per identificare gruppi di prodotti simili e gli esiti di processi di produzione, step fondamentali per il controllo della qualità, l'ottimizzazione dei processi e l'identificazione di potenziali problemi nella catena di produzione.

4.1.6 Focus sulle applicazioni mediche e biostatistiche

Inoltre, la cluster analysis applicata su dati clinici, genetici o di imaging [\[Kalyani 12\]](#) [\[Manipur 18\]](#) [\[Aliusef 22\]](#) [\[Godwin 18\]](#) sta assumendo grande importanza in ambito medico, per ottenere strategie di trattamento

più personalizzate, individuare clusters di pazienti particolarmente a rischio di insorgenza di determinate patologie, definire valori più significativi per l'insorgenza o meno di malattie o testare le differenze negli outcomes di terapie.

In particolare, ambiti di applicazione più specifici degli ultimi anni comprendono:

- **Clustering delle malattie per le strategie di trattamento:** la cluster analysis è ampiamente utilizzata nella ricerca medica per identificare sottotipi distinti di malattie; ad esempio, nella ricerca sul cancro, il clustering dei dati molecolari può rivelare sottotipi diversi con risposte varie ai trattamenti possibili; si tratta di informazioni cruciali per adattare le strategie di trattamento ai singoli pazienti, associandoli a clusters di pazienti con caratteristiche comuni.
- **Stratificazione dei pazienti per gli studi clinici:** la cluster analysis aiuta nella stratificazione dei pazienti, identificando sottogruppi con caratteristiche simili, in modo da aiutare i ricercatori a progettare trial più mirati ed efficaci; questo migliora le possibilità di rilevare gli effetti del trattamento in specifiche popolazioni di pazienti.
- **Identificazione dei profili biomolecolari:** la cluster analysis viene impiegata per identificare profili biomolecolari associati a specifiche malattie, raggruppando i pazienti in base ai loro profili molecolari o genetici, ed andando a scoprire pattern che possono fungere da biomarcatori per diagnosi, prognosi o risposta ai trattamenti.
- **Ricerche sulla salute mentale:** negli studi sulla salute mentale, la cluster analysis è utilizzata per identificare sottotipi di disturbi mentali o gruppi di pazienti con profili sintomatici simili; questo permette lo sviluppo di piani di trattamento personalizzati, ed aiuta a comprendere l'eterogeneità all'interno delle condizioni di salute mentale.
- **Studi epidemiologici:** la cluster analysis aiuta a identificare pattern di occorrenza e trasmissione delle malattie; dunque le informazioni ottenute dalla cluster analysis sono preziose per comprendere epidemie, progettare misure preventive e allocare risorse in modo efficiente ed efficace.
- **Predizione della risposta ai farmaci:** la cluster analysis viene utilizzata per predire le risposte individuali dei pazienti a specifici farmaci, raggruppando i pazienti con profili molecolari o clinici simili; in questo modo, i ricercatori possono inferire le potenziali risposte al trattamento, guidando approcci di medicina personalizzata.
- **Clustering dei registri sanitari elettronici (EHR):** nell'informatica sanitaria, la cluster analysis è utilizzata per raggruppare pazienti con registri sanitari simili, utile per rivelare pattern nascosti nei dati dei pazienti, informare le decisioni cliniche e migliorare la gestione sanitaria.
- **Analisi delle immagini funzionali del cervello:** in neuroscienze, la cluster analysis è applicata ai dati di imaging funzionale del cervello per identificare regioni con pattern di attivazione simili, aiutando a comprendere la funzione cerebrale, mappare le reti neurali e studiare disturbi neurologici complessi.
- **Modellizzazione delle malattie infettive:** la cluster analysis viene spesso impiegata nella modellizzazione della diffusione di malattie infettive; raggruppando regioni o popolazioni con dinamiche di trasmissione simili, i ricercatori possono sviluppare interventi mirati e strategie di contenimento efficaci.
- **Clustering delle varianti genetiche in genetica:** la cluster analysis è utilizzata per raggruppare individui in base alle loro varianti genetiche, facilitando l'identificazione di sottopopolazioni genetiche, essenziale per comprendere la genetica di popolazione, i modelli di migrazione e la suscettibilità alle malattie.

4.1.7 Integrazione con tecniche di riduzione della dimensionalità

Spesso la cluster analysis viene integrata con tecniche di riduzione della dimensionalità con fini di miglioramento della visualizzazione (i dati a dimensione ridotta sono più facili da visualizzare, facilitando l'interpretazione delle strutture dei clusters), miglioramento dell'efficienza computazionale (la riduzione della dimensionalità può accelerare il processo di clustering, specialmente per algoritmi sensibili a dati ad alta dimensionalità) e per la riduzione del rumore (focus sulle caratteristiche più rilevanti, in modo da mitigare l'impatto del rumore sui risultati della cluster analysis).

Tuttavia, la riduzione di dimensionalità comporta anche alcuni problemi di non semplice soluzione, come la perdita di informazioni (influenzando potenzialmente l'accuratezza della cluster analysis), la sensibilità degli algoritmi (la scelta del metodo di riduzione della dimensionalità è fondamentale, e richiede sempre una considerazione attenta) e la conseguente ottimizzazione dei parametri (essenziale per ottenere risultati validi).

4.2 K-means

4.2.1 Introduzione

Il clustering K-means (anche detto algoritmo di Lloyd) [\[Nielsen 16\]](#) [\[Dubey 17\]](#) è un algoritmo fondamentale di apprendimento automatico non supervisionato, progettato per suddividere un set di dati in gruppi distinti, basati sulle somiglianze tra i punti dati. Viene ampiamente utilizzato in vari campi, tra cui l'analisi dei dati, l'elaborazione delle immagini, la bioinformatica e il text mining.

L'algoritmo K-means, basato sul criterio della somma dei quadrati, opera in modo non supervisionato, non basandosi su dati di allenamento etichettati, bensì esplora invece la struttura intrinseca dei dati per formare clusters, con ogni cluster che rappresenta un gruppo distinto dei dati. L'algoritmo cerca di minimizzare la varianza all'interno dei clusters (detta anche *varianza within*) e massimizzare la varianza tra di essi (anche chiamata *varianza between*), cercando fondamentalmente di identificare clusters internamente compatti ed esternamente ben separati.

K-means trova applicazioni in diversi campi [\[Kumar 18\]](#) grazie alla sua versatilità e semplicità. Ecco sei esempi tipici di utilizzo della tecnica di cluster analysis:

- **Segmentazione dei Clienti:** aziende, imprese di e-commerce spesso utilizzano K-means come metodo per segmentare i clienti in base ai loro comportamenti d'acquisto; i clienti con modelli di acquisto simili vengono raggruppati, così che le aziende possano adottare strategie di marketing mirate, ottimizzare le raccomandazioni di prodotti e servizi e migliorare l'esperienza di acquisto del cliente.
- **Compressione delle Immagini:** nell'elaborazione delle immagini, K-means può essere applicato per ridurre il numero di colori in un'immagine; raggruppando colori simili e rappresentando ciascun cluster con il suo centroide, l'algoritmo permette di comprimere efficacemente l'immagine preservando la qualità visiva.
- **Rilevamento delle Anomalie:** nella sicurezza informatica, K-means può essere impiegato per il rilevamento delle anomalie per identificare modelli insoliti nel traffico di rete; raggruppando il comportamento medio di utenti online, qualsiasi deviazione da questi clusters può essere segnalata come potenziale minaccia alla sicurezza o anomalia.
- **Clustering dei Documenti:** tecniche di text mining ed elaborazione del linguaggio naturale spesso utilizzano K-means per il clustering di documenti, in modo tale da raggruppare documenti simili, col fine di un efficace recupero delle informazioni, di una semplificata modellazione dei topics all'interno di una collezione di documenti e di riassunti di corpora di documenti.
- **Analisi dei Dati Genomici:** in bioinformatica, K-means viene applicato per raggruppare geni in base ai loro modelli di espressione; identificare gruppi di geni con profili di espressione simili può fornire approfondimenti sui processi biologici, contribuendo in modo significativo alla comprensione dei meccanismi genetici.
- **Sistemi di raccomandazioni:** in diversi contesti, come le vendite online, la fruizione di servizi di intrattenimento (come film, serie tv, musica, libri...), l'algoritmo K-means può essere utilizzato per fornire proposte efficaci, basandosi sugli interessi di altri utenti con comportamenti ed interessi simili all'utente a cui sono indirizzati i suggerimenti.

4.2.2 Passaggi algoritmici

L'algoritmo K-means segue un processo iterativo [\[Boutsidis 15\]](#) per rinforzare iterativamente le assegnazioni dei clusters. I passaggi chiave sono i seguenti:

1. **Normalizzazione dei dati:** il metodo K-means è molto sensibile agli outliers, per questo motivo la fase di normalizzazione dei dati è fondamentale per ottenere risultati validi.

2. **Scelta del numero di clusters** (e dunque il numero di centroidi): si tratta di un problema fondamentale dell'algoritmo, in quanto il metodo non prevede una scelta automatica del numero di clusters in base a qualche indice o valore, bensì una scelta dell'utente per un numero K di clusters.
3. **Definizione dei centroidi**: i centroidi sono i punti definiti come centro di ogni cluster (dunque a K clusters corrispondono K centroidi); inizialmente viene scelto casualmente un numero di centroidi pari al numero di clusters ricercati, e i centroidi sono scelti casualmente all'interno dell'insieme di dati, non avendo ancora informazioni approfondite sui dati a disposizione.
4. **Assegnazione dei punti ai clusters**: ogni punto viene assegnato al cluster il cui centroide è più vicino in termini di distanza euclidea, calcolata nel seguente modo:

$$d_{ij} = \sqrt{\sum_{k=1}^n (x_{ik} - x_{jk})^2}$$

In questa fase, dunque, viene a formarsi un numero K di clusters pari al numero K di centroidi.

5. **Aggiornamento dei centroidi**: viene ricalcolata la media (dunque il centroide) di tutti i clusters utilizzando i punti assegnati al cluster stesso:

$$\mu_i = \frac{1}{|S_i|} \sum_{x \in S_i} x$$

6. **Ripetizione passaggi 3 e 4 fino alla convergenza dell'algoritmo**: la convergenza avviene quando:
 - i centroidi non cambiano più significativamente
 - viene raggiunto il numero massimo di iterazioni prestabilito all'inizio.
 - i punti non cambiano più il cluster a cui sono assegnati

Dunque gli ultimi centroidi, e quindi anche gli ultimi clusters trovati, sono il risultato finale ottenuto col metodo K-means applicato.

4.2.3 Iperparametri e tuning

L'algoritmo K-means coinvolge una serie di iperparametri [\[Ahmed 20\]](#) che influenzano significativamente le sue prestazioni ed i risultati ottenuti (i seguenti parametri riportati sono quelli modificabili con l'algoritmo *KMeans*, della libreria *sklearn*, utilizzato su *Python*):

- **n_clusters (default: 8)**: definisce il numero K di clusters che l'algoritmo mira a creare; si tratta del parametro più importante.
- **init (default: ‘kmeans++’)**: permette di definire il metodo con cui sono scelti i centroidi iniziali da cui l'algoritmo inizia ad eseguire la procedura:
 - ‘K-means++’: tecnica che seleziona un insieme di centroidi iniziali basato su una distribuzione empirica del contributo dei punti sull'inerzia complessiva dei punti; questo metodo permette di arrivare alla convergenza più velocemente.
 - ‘random’: seleziona i centroidi in maniera random tra i punti presenti
 - Può essere anche definito un array con dei punti (esistenti o creati artificialmente) da definire come centroidi iniziali.
- **n_init (default: ‘auto’)**: definisce il numero di volte che viene rieseguito l'algoritmo con un seed diverso; se ‘auto’, il valore dipende dal parametro “init”.
- **max_iter (default: 300)**: numero massimo di iterazioni che può eseguire l'algoritmo per arrivare alla convergenza.

- **tol (default: 0.0001)**: tolleranza relativa alla norma di Frobenius della differenza tra due centroidi in due iterazioni consecutive, per dichiarare la convergenza raggiunta.
- **algorithm (default: ‘lloyd’)**: tipo di algoritmo K-means da usare; il default è ‘lloyd’, altrimenti esiste l’opzione ‘elkan’ più indicata (ma anche più computazionalmente dispendiosa) per datasets con clusters ben definiti.

4.2.4 Il numero ottimale di clusters k : una scelta complessa

Determinare il valore ottimale di K , numero di clusters (e dunque di centroidi) è cruciale [Kapil 16] [Hamerly 03], e a tal proposito esistono una serie di tecniche con questo scopo.

Un metodo comune è il metodo del gomito (in inglese, “*elbow method*”): l’algoritmo viene eseguito con una serie di valori di K , e per ognuno di questi viene calcolata la WCSS (Within-Cluster Sum of Squares, ovvero la funzione obiettivo che si punta a minimizzare col clustering); il punto in cui la riduzione della varianza rallenta (formando una curva a gomito) viene considerato come il numero di clusters K ottimale.

Un secondo metodo consiste nel calcolare l’indice di Silhouette (indice che misura la qualità del clustering), e come nel caso del metodo del gomito, si ripete l’operazione per una serie di valori di K , in modo da trovare il valore di K per cui è massimo l’indice di Silhouette.

In generale, la letteratura e studi pregressi in ambito biomedico, riportano un utilizzo più ricorrente di metodo K-means applicato con un numero di clusters compreso tra due e cinque.

4.2.5 Vantaggi e svantaggi

Tra i vantaggi dell’algoritmo K-means abbiamo:

- **Semplicità ed efficienza**: l’algoritmo K-means è diretto e computazionalmente efficiente (a livello di requisiti di memoria, tempo richiesto e ridotta capacità computazionale richiesta dai dispositivi utilizzati per eseguirlo), rendendolo adatto per set di dati di grandi dimensioni.
- **Scalabilità**: l’algoritmo K-means si adatta bene alle dimensioni del set di dati, rendendolo applicabile a set di dati con un elevato numero di punti dati.
- **Interpretabilità**: i risultati ottenuti sono semplici da interpretare, poiché i clusters sono formati sulla base della media dei punti dati al loro interno.
- **Versatilità**: l’algoritmo K-means può essere applicato a svariate tipologie di dati, e non è limitato a distribuzioni specifiche dei dati.

Tuttavia, l’algoritmo K-means presenta comunque una serie di svantaggi, tra cui:

- **Sensibilità ai centroidi iniziali**: le prestazioni dell’algoritmo K-means possono essere molto sensibili alla selezione iniziale dei centroidi, portando potenzialmente a soluzioni subottimali in caso di scelte iniziali poco indicate.
- **Ottimi locali**: K-means può convergere a ottimi locali, e il risultato finale dipende dalle condizioni iniziali, influenzando la qualità del clustering (due centroidi iniziali molto vicini possono condizionare il risultato finale, non rappresentando realmente la corretta divisione dei dati in clusters).
- **Assunzione di clusters sferici**: l’algoritmo K-means assume che i clusters siano sferici e di dimensioni uguali, rendendolo meno efficace in alcuni contesti in cui i clusters sarebbero non convessi, o comunque di dimensioni irregolari.
- **Dipendenza dalla scelta di K clusters/centroidi**: l’algoritmo K-means richiede la specifica del numero di cluster, scelta non sempre semplice e scontata, specialmente in contesti in cui il numero ottimale non è noto a priori né di facile definizione.
- **Sensibilità agli outliers**: gli outliers possono influenzare significativamente il calcolo della media, e dunque il calcolo dei centroidi ad ogni aggiornamento nell’esecuzione dell’algoritmo, influenzando il

risultato complessivo del clustering. Per questo motivo, è fondamentale la fase di normalizzazione dei dati.

- **Convergenza lenta dei dati:** in caso di dataset molto grandi, la convergenza può essere più lenta, e dunque bisogna prestare particolare attenzione alla scelta di alcuni parametri, come il *max_iter*.

4.3 Clustering gerarchico

4.3.1 Introduzione

Il Clustering Gerarchico [Nielsen 16] [Bridges 66] è una potente tecnica nell'apprendimento automatico non supervisionato, progettata principalmente per organizzare i dati in una struttura gerarchica di clusters nidificati. A differenza dei metodi di partizionamento come il K-means, il clustering gerarchico crea una struttura a forma di albero, nota come dendrogramma, che rappresenta le relazioni tra i punti dati. Questa analisi completa mira a chiarire le complessità scientifiche del clustering gerarchico, approfondendone lo scopo, i parametri, le strategie di regolazione, le applicazioni, i passaggi algoritmici e i vantaggi e gli svantaggi associati.

Il principale scopo del clustering gerarchico è organizzare i dati in una gerarchia di clusters, consentendo un'esplorazione dettagliata delle somiglianze e delle differenze tra i dati. A differenza degli algoritmi di partizionamento, il clustering gerarchico fornisce una rappresentazione ricca della struttura dei dati, offrendo approfondimenti sulle relazioni globali e locali. L'algoritmo raggiunge questo obiettivo fondamentalmente unendo o dividendo iterativamente i clusters, formando un dendrogramma che racchiude l'organizzazione gerarchica dei dati.

Il clustering gerarchico si divide in due tipologie differenti: il clustering gerarchico agglomerativo (basato su un approccio *bottom-up*, in cui si inizia con clusters formati da una sola osservazione e uniti a due a due fino ad avere un unico cluster contenente tutti i dati) e il cluster gerarchico divisivo (con approccio *top-down*, in cui tutti gli elementi fanno parte di un unico cluster inizialmente, che viene diviso in sottoclusters ad ogni passaggio fino ad ottenere un cluster per ogni osservazione). Il clustering gerarchico agglomerativo è anche conosciuto come AGNES, o Agglomerative Nesting.

Il clustering gerarchico trova applicazioni in diversi campi [Papin 21] [Vaura 20] grazie alla sua capacità di rivelare relazioni gerarchiche nei dati. Ecco cinque esempi notevoli:

- **Tassonomia in biologia:** il clustering gerarchico può essere utilizzato per organizzare le specie in base alle somiglianze genetiche; utile per creare tassonomie, comprendere le relazioni evolutive e classificare le specie in base alle caratteristiche genetiche.
- **Segmentazione dei clienti nel marketing:** gli esperti di marketing utilizzano spesso il clustering gerarchico per suddividere i clienti in base a varie caratteristiche come comportamento d'acquisto, informazioni demografiche e preferenze; in questo modo, è facile personalizzare strategie di marketing per segmenti specifici di clienti, ottimizzando al massimo la pubblicità e le offerte personalizzate.
- **Classificazione dei documenti nell'elaborazione del linguaggio naturale:** il clustering gerarchico permette di organizzare un ampio corpus di documenti di testo in base alle somiglianze di contenuto e topics; aiuta nella classificazione dei documenti, nella modellazione dei temi e nella sintesi raggruppando documenti simili.
- **Analisi delle immagini in Artificial Vision:** il clustering gerarchico può essere impiegato per organizzare le immagini in base alle caratteristiche visive; aiuta nel recupero delle immagini basato sul contenuto, consentendo il recupero efficiente e rapido di immagini simili.
- **Analisi dei dati genomici in bioinformatica:** il clustering gerarchico è applicato per analizzare i profili di espressione genica in diverse condizioni; aiuta ad identificare gruppi di geni con modelli di espressione simili, fornendo approfondimenti su processi biologici e vie metaboliche.

4.3.2 Dendrogramma

Un dendrogramma [Murtagh 12] è un diagramma che mostra le relazioni gerarchiche tra osservazioni che fanno parte dello stesso set di dati. Si tratta di una tipologia di grafico strettamente collegata col clustering gerarchico, in quanto è il metodo più efficace di rappresentare l'andamento a fasi dell'algoritmo.

La chiave principale di lettura di un dendrogramma è relativa all'altezza in cui due oggetti vengono uniti: infatti è possibile vedere il dendrogramma e la sua struttura ad albero come se fosse un riassunto della matrice di distanze calcolabile sui dati di interesse.

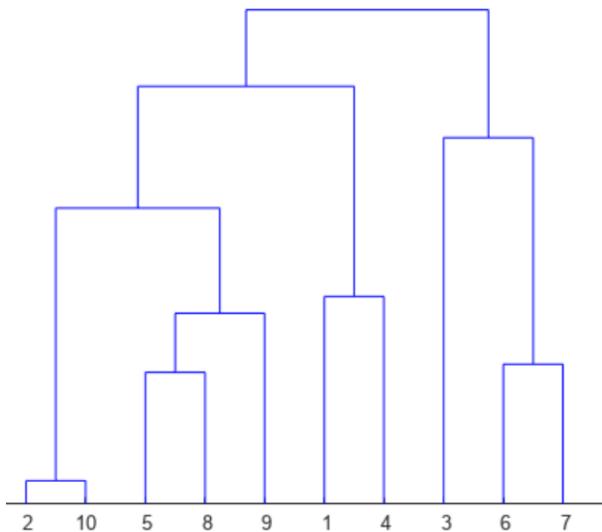


Figura 4.20: esempio di dendrogramma

Nella figura [4.20], si può notare che le osservazioni 2 e 10 sono le più simili (e quindi vicine nella matrice di distanze) nel dataset, motivo per cui l'unione in un solo cluster avviene nel primo passaggio in caso di clustering gerarchico agglomerativo. Al contrario, le osservazioni 3, 6 e 7 si uniscono a tutte le altre all'ultimo passaggio: questo significa che, volendo osservare un risultato con due clusters che comprendono tutti i dati, il cluster con le osservazioni 3,6,7 e il cluster comprendente tutte le altre osservazioni sarebbero i due clusters finali.

Un'operazione fondamentale è il cosiddetto “taglio del dendrogramma”, che consiste nel tagliare l'immagine in un punto definito da una serie di possibili parametri: il dendrogramma può essere tagliato in modo tale da ottenere una clusterizzazione con un numero preciso di clusters (ad esempio, se la necessità è avere un numero predefinito di clusters); oppure, il dendrogramma può essere tagliato secondo una serie di indici (come la minimizzazione del WCSS, ovvero la Within-Cluster Sum of Squares, ovvero la funzione obiettivo che si punta a minimizzare col clustering).

4.3.3 Passaggi algoritmici

Il clustering gerarchico agglomerativo segue un processo ricorsivo per costruire un dendrogramma. I passaggi chiave sono i seguenti:

1. **Normalizzazione dei dati:** il clustering gerarchico è molto sensibile agli outliers, per questo motivo la fase di normalizzazione dei dati è fondamentale per ottenere risultati validi.
2. **Inizializzazione:** in questa fase iniziale, ogni punto dati viene considerato come un cluster singolo.
3. **Calcolo delle distanze tra coppie:** viene calcolata la distanza tra ogni coppia di clusters utilizzando la metrica di distanza scelta, e i parametri definiti.

4. **Unione/Divisione:** una volta identificata la coppia di clusters con la distanza più piccola, in base al criterio di collegamento scelto, la coppia di clusters viene unita in un nuovo cluster.
5. **Aggiornamento della matrice delle distanze:** a questo punto viene ricalcolata la matrice di distanze, considerando il nuovo cluster appena creato e tutti gli altri clusters.
6. **Ripetizione passaggi dal 3 al 5:** vengono ripetuti iterativamente i passaggi 3, 4 e 5 fino a quando rimane un singolo cluster che comprende tutti i punti dati.

In parallelo al processo viene anche costruito il dendrogramma relativo al risultato della procedura di clustering.

4.3.4 Tipi di distanza

Ci sono diverse tipologie di distanza [\[Vark 04\]](#) [\[Lee 07\]](#) [\[Kim 16\]](#) [\[Kumar 14\]](#) [\[Grabusts 11\]](#) che possono essere utilizzate nell'algoritmo:

- **Distanza euclidea:** detta anche “distanza l2”, è calcolata come norma euclidea della differenza tra due punti (o unità statistiche):

$${}_2d_{ij} = \left\| x_i - x_j \right\| = \sqrt{\sum_{s=1}^p (x_{is} - x_{js})^2}$$

- **Distanza di Manhattan:** detta anche “distanza della città a blocchi”, o “metrica del taxi”, o “distanza l1”, è pari alla somma dei cateti che uniscono due punti (o unità statistiche), pensando a due cateti paralleli agli assi:

$${}_1d_{ij} = |x_{is} - x_{js}|$$

- **Distanza di Minkowski:** si tratta praticamente di una formula generale, da cui è possibile ricavare tutte le altre distanze precedentemente citate (con $k=2$ è la distanza euclidea, con $k=1$ è la distanza di Manhattan...)

$${}_kd_{ij} = \left[\sum_{s=1}^p |x_{is} - x_{js}|^k \right]^{\frac{1}{k}}$$

- **Distanza coseno:** misura quanto sono simili due vettori:

$$1 - \cos(\theta) = \frac{A \cdot B}{|A||B|} = \frac{\sum_{i=1}^n A_i B_i}{\sqrt{\sum_{i=1}^n A_i^2} \sqrt{\sum_{i=1}^n B_i^2}}$$

Inoltre, è importante ricordare alcune proprietà legate alla famiglia di distanze di Minkowski:

1. La metrica di Minkowski è una funzione non crescente dell'indice k , per cui valgono le seguenti diseguaglianze:

$${}_1d_{ij} \geq {}_2d_{ij} \geq \dots \geq {}_\infty d_{ij}$$

2. Le distanze di Minkowski sono invarianti per translazione delle variabili:

$${}_k d(x_i + c; x_j + c) = {}_k d(x_i; x_j)$$

dove c è un vettore p -dimensionale di costanti.

- 3. Le distanze di Minkowski sono invarianti per trasformazioni ortogonali (rotazioni) delle variabili:

$${}_2 d(Tx_i; Tx_j) = {}_2 d(x_i; x_j)$$

dove T è una matrice $p \times p$ tale che $T^T T = I$.

4.3.5 Iperparametri e tuning

Gli algoritmi di clustering agglomerativo coinvolgono una serie di iperparametri, che influenzano significativamente le loro prestazioni ed i risultati ottenuti (i seguenti parametri riportati sono quelli modificabili con l'algoritmo *AgglomerativeClustering*, della libreria *sklearn*, utilizzato su *Python*):

- **n_clusters (default: 2):** definisce il numero di clusters che l'algoritmo mira a creare. Dev'essere posto pari a 'None' se il parametro "distance_threshold" ha un valore definito.
- **linkage (default: 'ward'):** il criterio di collegamento utilizzato, che determina quale distanza utilizzare per dividere/unire i clusters; può essere:
 - '**ward**' (o metodo di ward), definito come la minimizzazione dell'incremento di varianza per i due clusters uniti:

$$f = \sum_{i=1}^k \sum_{j=1}^l \frac{d(x, y)}{|C_1| * |C_2|}$$

si tratta di un metodo utile con clusters che hanno molto rumore, in quanto tende a separare comunque bene i clusters ottenuti; tuttavia, si tratta di un metodo che tende a riprodurre clusters globulari.

- '**single**' (o metodo del legame singolo): definito come la distanza minima tra i punti in due clusters diversi:

$$f = \min(d(x, y))$$

si tratta di un metodo utile in caso di forme non ellittiche dei clusters, e funziona bene a patto che la distanza tra clusters non sia piccola; infatti, se c'è rumore tra i due clusters, la distanza singola tende a separare male clusters sovrapposti.

- '**average**' (o metodo del legame medio): definito come la distanza media tra i punti in clusters diversi:

$$f = \frac{1}{kl} \sum_{i=1}^k \sum_{j=1}^l d(x, y)$$

il metodo del legame medio tende a separare bene clusters che tendono a sovrapporsi, e in generale clusters con rumore; si tratta, tuttavia, di un metodo che solitamente riproduce clusters globulari.

- '**complete**' (o metodo del legame completo): definito come la distanza massima tra i punti in due clusters diversi:

$$f = \max(d(x, y))$$

il metodo del legame completo tende a separare bene clusters che tendono a sovrapporsi, e in generale clusters con molto rumore; tuttavia si tratta di un metodo che tende a “rompere” grandi clusters (dividendoli in due o più clusters), e che tende a riprodurre clusters globulari come il metodo del legame medio.

- **metric (default: ‘euclidean’)**: la metrica utilizzata per calcolare la distanza; di default, viene definita come ‘euclidean’ (uguale a ‘l2’), mentre altre opzioni utilizzabili sono ‘manhattan’ (o ‘l1’), ‘cosine’ o ‘precomputed’ (solo se viene data una matrice di distanze come metodo di input dell’algoritmo); inoltre, se il metodo di linkage scelto è ‘ward’, solo la metrica ‘euclidean’ è valida.
- **distance_threshold (default: ‘None’)**: definisce il limite che porta l’algoritmo alla convergenza, dunque a fermarsi ad un certo numero di clusters trovati; se pari a ‘None’, anche “n_clusters” dev’essere ‘None’.

4.3.6 Vantaggi e svantaggi

Tra i vantaggi [Murtagh 12] del clustering gerarchico abbiamo:

- **Rappresentazione gerarchica**: l’algoritmo fornisce una rappresentazione gerarchica delle relazioni tra i dati, consentendo un’esplorazione dettagliata delle strutture dei clusters.
- **Nessuna assunzione sulla forma dei clusters**: a differenza di alcuni algoritmi (ad esempio, il K-means), il clustering gerarchico non presuppone l’assunzione di una forma specifica da parte dei clusters, rendendolo molto più versatile.
- **Nessuna necessità di specificare il numero di clusters a priori**: il clustering gerarchico non richiede necessariamente la specifica preventiva del numero di cluster (come altri metodi come il K-means), poiché produce una struttura gerarchica.
- **Applicabilità a diverse metriche di distanza**: l’algoritmo può utilizzare diverse metriche di distanza in base alla natura dei dati.
- **Interpretabilità**: la struttura ad albero del dendrogramma è intuitiva e facilita l’interpretazione delle relazioni tra i dati.

Tuttavia, l’algoritmo presenta comunque una serie di svantaggi, tra cui:

- **Complessità computazionale**: a livello temporale e di complessità dell’esecuzione l’algoritmo può essere dispendioso dal punto di vista computazionale, specialmente per set di dati di grandi dimensioni, rendendolo meno efficiente rispetto ad altri metodi di partizionamento.
- **Sensibilità a rumore ed outliers**: il clustering gerarchico può essere molto sensibile a rumore ed outliers, influenzando facilmente la stabilità dei clusters ottenuti; per questo motivo, è fondamentale il passaggio di normalizzazione dei dati.
- **Irreversibilità delle unioni**: una volta che i clusters vengono uniti, il processo è irreversibile, e la scelta del punto di unione può influenzare significativamente il risultato finale (a differenza di un procedimento come il K-means, dove l’algoritmo rielabora ad ogni passaggio i risultati ottenuti nel precedente).
- **Difficoltà nel gestire set di dati voluminosi**: i requisiti computazionali e l’uso della memoria aumentano parecchio con le dimensioni del set di dati, rendendolo di difficile utilizzo per set di dati molto grandi.
- **Dipendenza dalla metrica di distanza**: la scelta della metrica di distanza può influenzare sensibilmente il risultato finale del clustering e la selezione di una metrica appropriata richiede conoscenze di dominio.

4.4 Density-Based Spatial Clustering of Applications with Noise (DBSCAN)

4.4.1 Introduzione

DBSCAN, acronimo di Density-Based Spatial Clustering of Applications with Noise [Khan 14], è un algoritmo di clustering robusto e ampiamente utilizzato nell'apprendimento automatico. Sviluppato da Martin Ester, Hans-Peter Kriegel, Jörg Sander e Xiaowei Xu nel 1996, l'algoritmo DBSCAN è particolarmente efficace nell'identificare clusters con forme variabili e nella gestione degli outliers all'interno dei dataset. A differenza degli algoritmi di clustering tradizionali, come il K-means, DBSCAN non assume un numero predefinito di cluster e può identificare clusters con forme irregolari. È particolarmente abile nel gestire clusters di dimensioni e forme diverse, rendendolo adatto a dataset in cui i clusters possono avere densità molto differenti.

Il concetto chiave di DBSCAN ruota attorno alla definizione di clusters come regioni dense di dati separate da regioni più sparse. Categorizza i punti come punti centrali, punti di bordo o punti di rumore, a seconda della loro densità nel dataset. L'algoritmo cerca di raggruppare i punti centrali insieme e identificare i clusters come regioni di elevata densità di punti.

Fondamentale dunque è la fase di regolazione di DBSCAN, che comporta la selezione di valori appropriati per alcuni parametri come epsilon (ϵ) e il numero minimo di punti (min_samples).

DBSCAN trova applicazioni in vari settori grazie alla sua flessibilità e capacità di gestire clusters di diverse forme e dimensioni. Ecco cinque esempi significativi:

- **Rilevamento di anomalie:** rilevamento di comportamenti anomali nel traffico di una rete; infatti, DBSCAN può identificare clusters in cui sono presenti comportamenti anomali e segnalare gli outliers come potenziali anomalie nei dati di rete.
- **Clustering di dati geospaziali:** clustering di posizioni geografiche in base alla densità dei punti, in modo da identificare clusters di dati basati sulla posizione, come punti caldi di criminalità, aree con elevata attività di clienti, aree molto trafficate...
- **Segmentazione di immagini:** suddivisione di immagini in regioni significative; DBSCAN può essere applicato per raggruppare pixel con caratteristiche simili, facilitando la segmentazione delle immagini per le applicazioni di artificial vision.
- **Sistemi di raccomandazione:** clustering di utenti in base alle loro preferenze e comportamenti; DBSCAN può identificare gruppi di utenti con preferenze e comportamenti simili, agevolando la creazione di sistemi di raccomandazione personalizzati.
- **Biologia e genomica:** clustering di geni basato sui modelli di espressione; utile per identificare gruppi di geni con modelli di espressione simili, in modo da fornire insights sui processi e le vie biologiche.

4.4.2 Passaggi algoritmici

DBSCAN segue un insieme diretto ma potente di fasi per identificare clusters e punti di rumore:

1. **Normalizzazione dei dati:** fase fondamentale per ottenere risultati validi, senza che questi vengano influenzati sensibilmente dalla presenza di outliers.
2. **Inizializzazione:** L'algoritmo procede a partire da un punto dati arbitrario.
3. **Identificazione dei punti centrali:** Viene determinato se il vicinato intorno al punto dati corrente contiene un quantitativo minimo di punti dati: in caso affermativo, il punto è contrassegnato come punto centrale.

4. **Espansione del cluster:** se viene trovato un punto centrale, l'algoritmo crea un nuovo cluster e aggiunge tutti i punti centrali collegati al cluster. Viene poi esplorato il vicinato di ogni punto centrale per trovare eventuali punti centrali aggiuntivi.
5. **Ripetizione passaggi 3 e 4:** fino a quando tutti i punti centrali e i loro punti raggiungibili sono assegnati a clusters, i passaggi 3 e 4 vengono ripetuti.
6. **Assegnazione dei punti di bordo:** vengono identificati i punti di bordo, ovvero quei punti raggiungibili da un punto centrale, ma che non sono abbastanza vicini per essere considerati punti centrali; questi punti di bordo sono comunque assegnati al cluster più vicino.
7. **Identificazione dei punti di rumore:** i punti rimanenti, non considerati né punti centrali né punti di bordo, vengono considerati come punti di rumore, poiché non fanno parte di alcun cluster.

4.4.3 Iperparametri e tuning

Il DBSCAN clustering utilizza alcuni iperparametri [Khan 14] che influenzano significativamente le prestazioni ed i risultati ottenibili (i seguenti parametri riportati sono quelli modificabili con l'algoritmo *DBSCAN*, della libreria *sklearn*, utilizzato su *Python*):

- **eps (default: 0.5):** definisce la massima distanza tra due campioni per far sì che essi possano essere considerati uno vicino dell'altro. Si tratta del parametro più importante, fondamentale per i risultati dell'algoritmo.
- **min_samples (default: 5):** si tratta del numero di campioni nel vicinato di un punto perché questo possa essere considerato “core point”.
- **algorithm (default: ‘auto’):** definisce il tipo di algoritmo utilizzato dal modulo NearestNeighbors per calcolare la distanza e trovare i vicini per ogni punto; può essere:
 - ‘auto’: l'algoritmo cerca di determinare il miglior approccio in base ai dati a disposizione.
 - ‘ball_tree’: miglioramento rispetto al ‘kd_tree’ per quanto riguarda il comportamento con dati di grandi dimensioni; questo metodo divide i dati in serie di iper-sfere annidate, rendendolo più costoso di un ‘kd_tree’ ma efficiente su dati di grandi dimensioni e molto strutturati.
 - ‘kd_tree’: miglioramento rispetto al ‘brute’, è una struttura ad albero binaria che divide le regioni dell'albero in regioni annidate, in modo da avere una costruzione dell'albero molto veloce; tuttavia, si tratta di un metodo molto efficiente con low-dimensional data ma che tende a diventare inefficiente con dataset con molte (>20) dimensioni.
 - ‘brute’: molto usata per piccoli dataset, poco efficiente e utile per grandi dataset.
- **metric (default: ‘euclidean’):** la metrica utilizzata per calcolare la distanza; di default, viene definita come ‘euclidean’ (uguale a ‘l2’), mentre altre opzioni utilizzabili sono ‘manhattan’ (o ‘l1’), ‘cosine’ o ‘precomputed’ (solo se viene data una matrice di distanze come metodo di input dell'algoritmo; inoltre, se il metodo di linkage scelto è ‘ward’, solo la metrica ‘euclidean’ è valida).
- **leaf_size (default: 30):** questo parametro indica la dimensione delle “foglie” per i tipi di albero indicati con il parametro ‘algorithm’;
- **p (default: ‘None’):** la potenza della distanza di Minkowski utilizzata per calcolare la distanza tra punti (se p viene posto uguale a ‘None’, pari al valore default, viene utilizzata la distanza euclidea);
- **n_jobs (default: ‘None’):** questo parametro indica il numero di operazioni parallele da eseguire (se n_{jobs} viene posto uguale a ‘None’, pari al valore default, viene utilizzata la distanza euclidea);

4.4.4 Vantaggi e svantaggi

Tra i vantaggi del DBSCAN [Khan 14] abbiamo:

- **Flessibilità nella forma dei clusters:** l'algoritmo DBSCAN può identificare clusters con forme arbitrarie, rendendolo adatto a dataset in cui i clusters hanno forme irregolari (a differenza di altri metodi come il K-means).

-
- **Rilevamento automatico degli outliers:** l'algoritmo permette di identificare ed etichettare automaticamente i punti di rumore come outliers, rendendolo robusto agli outliers nel dataset.
 - **Nessuna necessità di specificare il numero di cluster:** l'algoritmo DBSCAN, a differenza di altri metodi come il K-means, non richiede all'utente di specificare a priori il numero di cluster, poiché può adattarsi alla struttura di densità intrinseca dei dati.
 - **Gestisce facilmente clusters con diverse densità:** l'algoritmo DBSCAN è particolarmente efficace nell'identificare clusters con densità variabili, adattandosi a regioni con densità di punti dati sia alta che bassa.
 - **Robusto alle variazioni dei parametri:** DBSCAN è relativamente robusto alle variazioni dei parametri, dunque piccoli cambiamenti in *epsilon* o *min_samples* spesso non portano a cambiamenti molto significativi nei risultati.

Tuttavia, l'algoritmo presenta comunque una serie di svantaggi, tra cui:

- **Sensibilità alle impostazioni dei parametri:** le prestazioni dell'algoritmo possono essere molto sensibili alla scelta dei parametri e trovare valori adatti può richiedere conoscenze di dominio o sperimentazione.
- **Difficoltà con dati ad alta dimensionalità:** in spazi ad alta dimensionalità, il concetto di distanza utilizzato dall'algoritmo diventa meno intuitivo e l'efficacia del clustering basato sulla densità può diminuire sensibilmente.
- **Particolare sensibilità ai punti di bordo:** l'assegnazione dei punti di bordo può essere sensibile a piccoli cambiamenti nel dataset, portando a grandi variazioni potenziali nelle assegnazioni dei clusters.
- **Intensivo in memoria:** conservare la matrice delle distanze di raggiungibilità per tutti i punti dati può essere molto dispendioso in termini di memoria, rendendo DBSCAN meno adatto per dataset molto grandi e computazionalmente pesante.

4.5 BIRCH Clustering

4.5.1 Introduzione

BIRCH, acronimo di Balanced Iterative Reducing and Clustering using Hierarchies [Zhang 97] [Zhang 96] [Lorbeer 17], è un algoritmo di clustering progettato per set di dati di grandi dimensioni. Sviluppato da Tian Zhang, Raghu Ramakrishnan e Miron Livny nel 1996, BIRCH è particolarmente adatto per situazioni in cui l'intero dataset non può essere contenuto in memoria, rendendolo una scelta efficiente e scalabile per il clustering di dati. È progettato specificamente per gestire set di dati troppo estesi per adattarsi in memoria, rendendolo adatto a scenari in cui gli algoritmi di clustering tradizionali potrebbero incontrare limitazioni.

BIRCH si concentra sulla creazione di una rappresentazione compatta del dataset, chiamata albero di feature di clustering (definito CFT, o Clustering Feature Tree), che consente un clustering efficiente e minimizza la necessità di memorizzare l'intero dataset in memoria. Infatti, si tratta di una struttura ad albero, dove le foglie non sono i singoli dati, bensì sottoclusters di dati, in modo da avere benefici ed enormi risparmi in termini di memoria impiegata.

BIRCH impiega un processo in due fasi: innanzitutto, costruisce un albero CFT mediante la fusione e la compressione iterativa dei punti dati e, successivamente, applica un algoritmo di clustering tradizionale alla rappresentazione condensata per formare i clusters finali. Questo approccio consente a BIRCH di scalare bene con grandi dataset e fornisce un equilibrio tra l'uso della memoria e l'accuratezza del clustering. Inoltre, in base alla memoria disponibile, è fondamentale regolare il *branching_factor* e la soglia per ottenere un equilibrio tra una rappresentazione compatta e l'accuratezza del clustering.

Il BIRCH clustering trova applicazioni [Zhang 97] in vari settori, specialmente dove è necessario elaborare efficientemente set di dati di grandi dimensioni. Ecco alcuni esempi significativi:

- **Rilevamento di intrusione di rete:** nell'analisi dei dati del traffico di rete per identificare pattern di intrusione, l'algoritmo BIRCH può effettuare un clustering efficiente dei dati di traffico di rete per rilevare pattern insoliti, in modo da contribuire all'identificazione precoce di potenziali intrusioni.
- **Segmentazione di clienti nell'e-commerce:** l'algoritmo è utile per operare del clustering sui clienti, in base al loro comportamento d'acquisto e alle preferenze; BIRCH può gestire grandi dataset di transazioni, consentendo alle imprese di e-commerce di segmentare i clienti per marketing mirato e raccomandazioni personalizzate.
- **Clustering di documenti nel text mining:** l'algoritmo viene utilizzato per fare del clustering su grandi corpora di documenti per il topic modeling; BIRCH può infatti elaborare ed eseguire il clustering efficiente di dati testuali, contribuendo all'organizzazione dei documenti, alla sintesi di corpora di documenti e alla scoperta dei topic più ricorrenti.
- **Analisi di immagini in Computer Vision:** l'algoritmo BIRCH permette di clusterizzare features di immagini per il riconoscimento degli oggetti, grazie alla grande scalabilità che lo rende adatto per elaborare grandi dataset di immagini, contribuendo a compiti come la segmentazione e la ricerca di immagini basate sul contenuto.
- **Analisi di dati genomici:** il BIRCH può gestire vaste quantità di dati genomici, contribuendo all'identificazione di gruppi di geni con pattern di espressione simili e contribuendo alla comprensione dei processi biologici.

4.5.2 Passaggi algoritmici

Il BIRCH clustering segue un processo [Zhang 97] [Zhang 96] diviso in due fasi fondamentali per creare una rappresentazione compatta del dataset ed eseguire l'operazione di clustering:

1. **Costruzione dell'albero CFT:**

- **Normalizzazione dei dati:** fase fondamentale dell'algoritmo, per ottenere risultati validi e non influenzati dall'eventuale presenza di outliers.
- **Inizializzazione:** creazione iniziale di un albero CFT vuoto.
- **Inserimento dati:** iterazione attraverso i punti dati, col fine di inserire ciascun punto nell'albero CFT creato al passaggio precedente: se un nodo foglia può contenere il punto senza superare la soglia, inseriscilo nel nodo; se la soglia viene superata, suddividi il nodo foglia e ridistribuisci i punti dati.
- **Fusione:** viene eseguita la fusione iterativa dei nodi nell'albero CFT, per mantenere una rappresentazione compatta.

2. Applicazione dell'algoritmo di clustering:

- **Clustering su nodi dell'albero CFT:** dopo aver costruito l'albero CFT, viene applicato un algoritmo di clustering tradizionale (ad esempio, K-means) sui nodi foglia dell'albero CFT.
- **Risultati ed analisi:** i clusters risultanti vengono utilizzati per analisi e interpretazione.

4.5.3 Iperparametri e tuning

BIRCH coinvolge diversi parametri [\[Zhang 96\]](#) [\[Ramadhani 20\]](#) che possono essere regolati per ottimizzarne le prestazioni. Ecco i principali parametri e le strategie per la fase di tuning:

- **threshold (default: 0.5):** definita anche soglia, si tratta della soglia sotto la quale avviene o meno l'unione di un punto ed un sottocampione (in caso contrario, viene creato un nuovo sottocampione); valori molto piccoli comportano un numero superiore di sottoclusters, viceversa valori elevati portano alla creazione di sottoclusters più grandi.
- **branching_factor (default: 50):** parametro che definisce il numero massimo di CFT sottoclusters in ogni nodo (se viene superato, il nodo viene diviso in due col sottocluster redistribuito).
- **n_clusters (default: 3):** definisce il numero di clusters dopo l'ultimo passaggio dell'algoritmo; può essere ‘None’ se non si vuole ottenere un numero preciso di cluster (a partire dai subclusters creati dalla procedura) ma solamente ottenere i sottoclusters prodotti dall'algoritmo.

4.5.4 Vantaggi e svantaggi

Tra i vantaggi del BIRCH [\[Zhang 96\]](#) abbiamo:

- **Scalabilità ed uso della memoria:** l'algoritmo BIRCH è progettato per la scalabilità, grazie alla sua struttura gerarchica, rendendolo particolarmente adatto a grandi dataset che non possono essere contenuti in memoria.
- **Efficienza:** creando una rappresentazione compatta (ovvero l'albero CFT) del dataset, l'algoritmo BIRCH riduce al minimo la necessità di memorizzare l'intero dataset in memoria, migliorandone l'efficienza in termini di velocità di esecuzione e pesantezza computazionale.
- **Adattabilità:** BIRCH è adattabile a diversi tipi di algoritmi di clustering, consentendo agli utenti di scegliere un metodo adatto per il compito di clustering specifico.
- **Gestione di dati rumorosi:** il clustering BIRCH è robusto ai dati rumorosi, e la sua struttura gerarchica aiuta a catturare i modelli complessivi nei dati.

Tuttavia, l'algoritmo BIRCH presenta anche una serie di svantaggi, tra cui:

- **Sensibilità ai parametri:** le prestazioni dell'algoritmo BIRCH possono essere particolarmente sensibili alla scelta dei parametri, come il branched_factor e la threshold, richiedendo una fase di tuning attenta e precisa.
- **Limitato a clusters sferici:** il BIRCH è solitamente meno efficace nel trattare clusters con forme non sferiche, poiché assume clusters sferici durante la fase di clustering (simile al K-means).

- **Dipendenza dall'inizializzazione:** la qualità dei risultati del clustering può dipendere fortemente dall'inizializzazione dell'albero CFT e dalla scelta dell'algoritmo di clustering applicato nella seconda fase.
- **Complessità dell'implementazione:** implementare l'algoritmo BIRCH può essere più complesso rispetto ad altri algoritmi di clustering, richiedendo una comprensione più approfondita per non cadere in errori dovuti a mancanza di dimestichezza col metodo.
- **Limitato a certi tipi di dati e dataset:** sebbene sia efficiente per dataset di grandi dimensioni, l'algoritmo BIRCH potrebbe non essere la scelta ottimale per dataset con clusters ben separati e distinti.

4.6 Spectral Clustering

4.6.1 Introduzione

Lo Spectral Clustering è una tecnica potente nell'ambito dell'apprendimento automatico che sfrutta le proprietà spettrali dei dati per eseguire il clustering. A differenza dei metodi di clustering tradizionali che operano nello spazio di input, lo Spectral Clustering trasforma i dati in uno spazio differente, utilizzando gli autovettori di una matrice di similarità, ed esegue il clustering in questo spazio trasformato.

Lo Spectral Clustering è particolarmente efficace in scenari in cui i dati presentano strutture complesse, dunque dove gli algoritmi di clustering tradizionali potrebbero andare in difficoltà. Lo Spectral Clustering riesce a superare i limiti degli algoritmi di clustering “tradizionali” catturando la struttura sottostante dei dati attraverso la loro rappresentazione spettrale, consentendo assegnazioni di clusters più flessibili e “fuzzy”.

L'idea fondamentale dell'algoritmo ruota attorno alla rappresentazione dei dati come un grafo di similarità, in cui i nodi corrispondono ai punti dati e gli archi indicano un peso di similarità tra le coppie di punti che uniscono. Analizzando le proprietà spettrali di questo grafo, lo Spectral Clustering svela schemi e strutture nascoste, facilitando la formazione di clusters in uno spazio trasformato.

Lo Spectral Clustering trova applicazioni in vari settori grazie alla sua capacità di gestire strutture complesse e relazioni non lineari. Ecco cinque esempi notevoli:

- **Segmentazione di immagini in Computer Vision:** lo Spectral Clustering può essere utilizzato per suddividere un'immagine in regioni semanticamente significative; infatti, si tratta di una tecnica utile per rappresentare le similarità tra pixel, consentendo la segmentazione delle immagini in regioni coerenti basate su colore, texture o altre caratteristiche visive.
- **Analisi di reti sociali:** per identificare comunità o gruppi all'interno di una rete sociale, in quanto lo Spectral Clustering può rivelare strutture nascoste nelle reti sociali catturando i modelli di connettività tra gli utenti, contribuendo alla rilevazione delle comunità o alla pubblicità mirata.
- **Clustering di dati genomici in bioinformatica:** clustering di geni basato sui profili di espressione, in modo da identificare gruppi di geni con pattern di espressione simili, contribuendo alla comprensione dei meccanismi di regolazione genetica e delle vie biologiche.
- **Clustering di documenti in Natural Language Processing (NLP):** raggruppamento di documenti basato su similarità semantica; infatti, lo Spectral Clustering può elaborare matrici documento-termine o altre rappresentazioni per identificare clusters tematici in grandi collezioni di documenti, facilitando la modellizzazione dei topics e l'organizzazione documentale.
- **Rilevamento di anomalie in sistemi di rilevamento intrusione:** utile per rilevare comportamenti insoliti nel traffico di rete, ovvero punti dati che si discostano dai modelli normali nel traffico di rete, contribuendo alla rilevazione precoce di potenziali minacce alla sicurezza.

4.6.2 Passaggi algoritmo

Lo Spectral Clustering segue una serie di passaggi per identificare i clusters col suo algoritmo:

1. **Normalizzazione dei dati:** passaggio fondamentale per ottenere risultati validi e significativi.
2. **Costruzione del grafo di similarità:** dato un dataset con n punti dati, viene costruito un grafo di similarità in cui ogni nodo rappresenta un punto dati e gli archi codificano le similarità tra coppie di punti. Misure di similarità comuni includono il kernel gaussiano (RBF) o i k-NN (o k-Nearest Neighbors).

3. **Calcolo della matrice di affinità:** in base alla misura di similarità scelta, viene calcolata una matrice di affinità che cattura le relazioni tra i punti dati; questa matrice riflette la forza delle connessioni tra i punti nel grafo di similarità.
4. **Matrice dei gradi e matrice laplaciana:** viene poi calcolata la matrice dei pesi, che contiene informazioni sul peso totale degli archi di ciascun nodo nel grafo di similarità
5. **Autodecomposizione:** viene dunque eseguita l'autodecomposizione (o eigendecomposition) della matrice laplaciana, in modo tale da ottenere i suoi autovalori e gli autovettori corrispondenti; gli autovettori vengono ordinati in base agli autovalori corrispondenti.
6. **Riduzione di dimensionalità:** successivamente vengono selezionati i k autovettori corrispondenti ai k più piccoli autovalori, in modo tale da formare una nuova matrice e da ridurre efficacemente la dimensionalità dei dati a k dimensioni.
7. **Clustering nella dimensione ridotta:** un algoritmo di clustering tradizionale, come il K-means, viene applicato allo spazio ridotto formato dagli autovettori selezionati; il numero di clusters è determinato dal parametro k. Ciascun punto dati viene assegnato dunque ad uno dei clusters identificati in base alla sua rappresentazione nello spazio ridotto.

4.6.3 Iperparametri e tuning

L'algoritmo Spectral Clustering coinvolge una serie di iperparametri che influenzano significativamente le sue prestazioni e i risultati ottenuti (i seguenti parametri riportati sono quelli modificabili con l'algoritmo *SpectralClustering*, della libreria *sklearn*, utilizzato su *Python*):

- **n_clusters (default: 8):** definisce il numero di clusters ottenuti e la dimensione del sottospazio di proiezione.
- **eigen_solver (default: ‘None’):** definisce la strategia dell'autodecomposizione (se ‘None’, viene usata ‘arpack’, mentre altre possibili decomposizioni sono ‘lobpcg’ e ‘amg’).
- **n_components (default: ‘None’):** definisce il numero di vettori da usare per la decomposizione spettrale (se ‘None’, viene posta uguale a *n_clusters*).
- **n_init (default: 10):** numero di volte che l'algoritmo K-means viene fatto ripetere con centroidi differenti; il risultato finale sarà il miglior risultato ottenuto in termini di inerzia.
- **gamma (default: 1.0):** definisce il coefficiente del kernel rbf, poly, sigmoide, laplaciano e chi².
- **affinity (default: ‘rbf’):** indica come viene costruita la matrice di affinità (possibili scelte sono ‘nearest_neighbors’, ‘rbf’, ‘precomputed’, ‘precomputed_nearest_neighbors’)
- **n_neighbors (default: 10):** numero di vicini da usare per costruire la matrice di affinità nel caso di affinity uguale a ‘nearest_neighbors’.
- **eigen_tol (default: ‘auto’):** definisce il criterio di stop per la decomposizione della matrice Laplaciana (se ‘auto’ dipende dal parametro *eigen_solver*).
- **degree (default: 3):** grado del kernel polinomiale (viene ignorato in caso di altri kernel).
- **n_jobs (default: ‘None’):** questo parametro indica il numero di operazioni parallele da eseguire (se ‘None’, ovvero il default, è posta uguale a 1, -1 invece indica tutti i processori);

4.6.4 Vantaggi e svantaggi

Tra i vantaggi dello Spectral Clustering abbiamo:

- **Flessibilità nelle forme dei clusters:** lo Spectral Clustering può identificare clusters con forme arbitrarie, rendendola adatta a dataset con strutture complesse.
- **Gestione delle relazioni non lineari:** l'algoritmo è in grado di catturare efficacemente relazioni non lineari e può identificare clusters che i metodi tradizionali potrebbero trascurare.
- **Efficacia in spazi ad alta dimensionalità:** lo Spectral Clustering può gestire dati ad alta dimensionalità, operando in uno spazio spettrale a dimensionalità ridotta.

-
- **Capacità di catturare la struttura globale:** esaminando le relazioni globali tra i punti dati, lo Spectral Clustering è efficace nel catturare la struttura complessiva dei dati.
 - **Meno sensibile all'inizializzazione:** lo Spectral Clustering è meno sensibile all'inizializzazione rispetto ad alcuni algoritmi di clustering tradizionali (come il K-means), contribuendo alla sua stabilità.

Tuttavia, l'algoritmo Spectral Clustering presenta comunque una serie di svantaggi, tra cui:

- **Sensibilità al parametro k :** la scelta del numero di cluster (k) può influenzare significativamente i risultati del clustering e la selezione di un valore inappropriato può portare a partizioni subottimali.
- **Scalabilità:** a causa del calcolo dell'autodecomposizione, lo Spectral Clustering potrebbe risultare particolarmente dispendioso dal punto di vista computazionale e dei tempi di esecuzione, soprattutto per dataset molto grandi e con numero elevato di features.
- **Dipendenza dalle misure di similarità:** le prestazioni dello Spectral Clustering sono influenzate significativamente dalla scelta delle misure di similarità e dei metodi di costruzione della matrice di affinità, richiedendo conoscenze di dominio.
- **Limitazione a rumore basso e moderato:** lo Spectral Clustering potrebbe avere difficoltà con dataset che contengono livelli elevati di rumore, poiché presume un certo livello di coerenza tra i punti dati all'interno dei clusters e rischia di essere particolarmente sensibile in presenza di outliers.

4.7 Metodi di valutazione delle tecniche di cluster analysis

Valutare le prestazioni di un algoritmo di clustering non è semplice come contare il numero di errori, o calcolare la precisione e il recall di un algoritmo di classificazione supervisionata. In particolare, qualsiasi metrica di valutazione non dovrebbe tener conto dei valori assoluti delle etichette dei clusters, ma piuttosto se questo clustering definisce separazioni dei dati simili a un insieme di classi di riferimento, o soddisfa alcune assunzioni, come ad esempio che i membri appartenenti alla stessa classe siano più simili tra loro rispetto ai membri di classi differenti, il tutto in base ad una metrica di similarità prestabilita.

Per questo motivo, sono stati usati cinque metodi di valutazione [\[Kim 16\]](#) [\[Amigo 08\]](#) [\[Palacio-Nino 19\]](#) [\[Kumar 14\]](#) delle differenti tecniche di cluster analysis, che non prevedono la conoscenza della classe di appartenenza di ogni osservazione:

- **Il coefficiente di Silhouette:** si tratta di una metrica [\[Godwin 18\]](#) utilizzata per misurare la qualità dell'assegnazione dei punti ai corretti clusters in un dataset, combinando sia informazioni relative alla coesione (quanto un dataset è vicino ai punti con caratteristiche vicine) sia riguardo la separazione (quanto lontani sono i punti da altri che fanno parte di altri clusters, e che hanno dunque caratteristiche molto differenti); l'indice viene calcolato con la seguente formula

$$s = \frac{b - a}{\max(a, b)}$$

con a pari alla distanza media tra un campione (o centroide) e tutti gli altri punti nella stessa classe, e b che corrisponde alla distanza media tra un campione e tutti gli altri punti nel cluster più vicino successivo.

Il coefficiente di Silhouette assume un punteggio compreso tra -1 (valore minimo, per clustering non corretto) e 1 (valore massimo, corrisponde ad un clustering efficace, con gruppi molto densi e ben separati gli uni dagli altri), con valori vicini allo 0 in caso di clusters che tendono a sovrapporsi e non sono dunque ben separati.

- **L'indice di Calinski-Harabasz:** noto anche come Criterio del Rapporto di Varianza, è un indice simile all'F test e al test per l'ANOVA, e calcola il rapporto tra la somma della dispersione tra i clusters e la dispersione all'interno dei clusters per tutti i gruppi creati (dove la dispersione è definita come la somma delle distanze al quadrato). La formula è la seguente:

$$F = \frac{SS_{between}/(k - 1)}{SS_{within}/(n - k)}$$

con SS come la somma dei residui al quadrato (con distinzione di $SS_{between}$, ovvero tra gruppi, ed SS_{within} , ovvero entro i gruppi).

Si ricercano valori molto elevati, sinonimo di clusters densi e ben separati gli uni dagli altri, e l'indice assume valori compresi tra 0 (minimo, sinonimo di $SS_{between}$ molto elevati, ovvero il caso di clusters poco separati tra di loro, che tendono a fondersi) e $+\infty$ (valori ricercati, contraddistingue clusters ben distinti e molto densi internamente).

- **L'indice di Davies-Bouldin (o DBI):** si tratta di un indice che rappresenta la "similitudine" media tra i clusters, dove la similitudine è una misura che confronta la distanza tra i clusters con la dimensione stessa dei clusters; valori più bassi (con valore minimo 0) sono correlati ad un modello con una migliore separazione tra i clusters, mentre al crescere del valore dell'indice i clusters tendono ad essere separati peggio e meno densi internamente; la formula che definisce l'indice è la seguente:

$$DB \text{ index} = \frac{1}{k} \sum \max \left(\frac{(\Delta(X_i) + \Delta(X_j))}{\delta(X_i, X_j)} \right)$$

con $\Delta(X_i)$ come distanza interna di un cluster (ovvero la distanza di ogni punto di un cluster dal suo centroide), e $\delta(X_i, X_j)$ come la distanza intercluster tra i clusters X_i e X_j .

L'indice ha valori compresi tra 0 e $+\infty$. Si tratta di un indice utile perché non fa assunzioni particolari sulla forma dei clusters, ma che tuttavia risente particolarmente della presenza di outliers e rumore nei dati, e che può riportare errori non effettivamente presenti in questi casi.

- **L'indice di Dunn:** metodo utilizzato per valutare la qualità del clustering, viene calcolato come il rapporto tra la minore distanza tra due centroidi della distribuzione di dati, e la massima distanza tra due punti random di due clusters differenti; si tratta di un indice, dunque, che permette di valutare non solo quanto i clusters abbiano osservazioni con caratteristiche simili al loro interno, ma anche quanto siano differenti tra di loro i clusters. L'indice di Dunn assume valori compresi tra 0 e $+\infty$ (valore ricercato), e può essere calcolato nel seguente modo:

$$\text{Dunn index} = \frac{\min (dist(c_i, c_j))}{\max (dist(x_i, x_j))}$$

Non sempre è stato possibile calcolare questo indice, in quanto è molto dispendioso dal punto di vista computazionale, considerando la matrice di distanze per il calcolo dell'indice ed avendo dati a grandi dimensionalità.

- **L'entropy:** indice utilizzato per valutare la qualità del clustering, indicando come sono distribuiti i membri della distribuzione nei k clusters ottenuti; data l'entropia di un singolo cluster

$$\text{cluster entropy } E_C = \sum_{i=1}^k P(c_i) \log(P(c_i))$$

con $P(c_i)$ come probabilità di trovare un elemento all'interno del cluster con centroide c_i , è possibile calcolare l'entropia globale della clusterizzazione come

$$E = - \sum_j \frac{n_j}{n} \sum_{i=1}^k P(c_i c_j) \log_2(P(c_i c_j))$$

con $P(c_i c_j)$ come probabilità di trovare un elemento del cluster i nel cluster j , n_j come il numero di elementi nel cluster j ed n come numero totale di elementi nella distribuzione. L'entropia assume valori compresi tra 0 e $\log_2(k)$ (in caso di clusters più piccoli e tendenzialmente più numerosi), con K pari al numero di clusters.

4.8 Risultati e discussione cluster analysis

Sono state applicate le diverse tecniche di cluster analysis descritte nella sezione precedente. In particolare, è stato utilizzato il dataset finale ottenuto come risultato di una serie di operazioni di pulizia, come descritto nella sezione della riduzione di dimensionalità; vengono dunque analizzati dati formati da 272114 osservazioni, con 25 features qualitative e 44 variabili quantitative per un totale di 69. Su questo dataset verranno in seguito applicate le tecniche di cluster analysis ai dati su cui è stata effettuata riduzione di dimensionalità.

Inoltre, alcune tecniche (tra cui il clustering gerarchico agglomerativo, il DBSCAN e lo Spectral Clustering), erano computazionalmente dispendiose, in termini di requisiti dei dispositivi e di tempi di esecuzione: per questo motivo, per rendere i dati analizzabili e confrontabili tra i differenti metodi, è stato eseguito un campionamento, per scegliere 50000 osservazioni all'interno del dataset che si potessero confrontare tra le varie metodologie.

Sono stati riportati i risultati ottenuti con una serie di valori scelti dei parametri dei differenti metodi, ma per questi risultati sono state necessarie diverse verifiche sui valori più rilevanti (alcuni risultati non significativi non sono infatti stati riportati).

4.8.1 Clustering gerarchico agglomerativo

Osserviamo i risultati ottenuti col clustering gerarchico [\[scikit-learn.org\]](https://scikit-learn.org): sono state eseguite una serie di implementazioni sia con diverso *metodo di linkage* (metodo di *Ward*, *legame singolo*, *legame medio* e *legame completo*) sia con numeri differenti di clusters previsti (2,3,4,5,10,20,50 e 100), sia cambiando la *metrica* utilizzata (*euclidean* come standard, ma anche *l2*, *manhattan* e *coseno*).

Molti dei risultati ottenuti riportavano una divisione delle osservazioni in un grosso cluster formato dalla quasi totalità delle osservazioni, e una serie di piccolissimi clusters formati da poche decine di osservazioni; solamente col metodo di Ward, invece, si potevano osservare clusters di dimensioni più consistenti tra di loro.

Osserviamo prima di tutto il clustering col *metodo di Ward* (con uno dei codici eseguiti in *Python*, in questo caso per il calcolo del primo risultato con due clusters):

	Silhouette coefficient [-1,1]	Calinski-Harabasz index [0, +∞]	Davies-Bouldin Dunn index [0, +∞]	Entropy [0, log2(K)]	Biggest cluster dimension
2 clusters, dist. euclidea	0.242	6150.417	1.885	0.0059	9.209
3 clusters, dist. Euclidea	0.134	5342.179	2.097	0.0059	10.559
4 clusters, dist. Euclidea	0.133	4368.948	1.612	0.0059	9.852
5 clusters, dist. Euclidea	0.134	3915.061	1.446	0.0064	10.542
10 clusters, dist. Euclidea	0.064	2998.149	1.697	0.0073	10.446
20 clusters, dist. Euclidea	0.073	2621.090	1.093	0.0112	10.103
50 clusters, dist. Euclidea	0.013	1705.317	1.619	0.0213	10.458
100 clusters, dist. euclidea	-0.003	1094.753	1.889	0.0265	10.511

```
model_aggcl_2cl_ward_eucl = AgglomerativeClustering(n_clusters=2, linkage='ward', metric='euclidean')
```

Table 4.12: clustering gerarchico agglomerativo, metodo di Ward

I clusters ottenuti dalle diverse implementazioni del clustering gerarchico agglomerativo col metodo di Ward restituiscono come risultato diversi gruppi composti da un numero significativo di osservazioni: come è possibile osservare con le prossime tabelle, questi primi risultati ottenuti hanno un comportamento differente rispetto alle altre implementazioni del clustering gerarchico agglomerativo (solitamente con un unico grosso cluster, e altri piccolissimi gruppi di osservazioni).

Dal punto di vista delle metriche, i valori dell'indice di Silhouette sono tendenzialmente intorno allo 0, sinonimo di cluster che tendono a sovrapporsi; i valori dell'indice di Calinski-Harabasz sono tendenzialmente elevati, sinonimo di clusters separati ed internamente densi, e tendono a calare all'aumentare del numero di cluster; i valori per l'indice di Davies-Bouldin sono invece bassi, ad indicare buona separazione dei clusters; l'indice di Dunn presenta valori vicino allo 0, sinonimo di clustering non ottimale; infine, i valori dell'entropia sono tendenzialmente elevati, e rimangono costanti al variare del numero di cluster.

Ora osserviamo il clustering col metodo del legame medio (*average linkage*), con uno dei codici eseguiti in *Python*:

	Silhouette coefficient [-1,1]	Calinski-Harabasz index [0, +∞]	Davies-Bouldin Index [0, +∞]	Dunn Index [0, +∞]	Entropy [0, log2(K)]	Biggest cluster dimension
2 clusters, dist. euclidea	0.960	822.033	0.027	0.5932	0.0	49999
3 clusters, dist. Euclidea	0.960	829.743	0.027	0.5982	0.637	49998
4 clusters, dist. Euclidea	0.959	1135.278	0.128	0.3215	1.277	49996
5 clusters, dist. Euclidea	0.958	1264.303	0.130	0.3793	1.657	49994
10 clusters, dist. Euclidea	0.933	884.279	0.293	0.2199	10.819	49988
20 clusters, dist. Euclidea	0.886	784.039	0.341	0.1705	3.7257	49949
50 clusters, dist. Euclidea	0.780	494.557	0.274	0.1891	4.4893	49873
100 clusters, dist. euclidea	0.523	286.634	0.428	0.1319	10.743	49676

```
model_aggcl_2cl_avg_eucl = AgglomerativeClustering(n_clusters=2, linkage='average',
metric='euclidean')
```

Table 4.13: clustering gerarchico agglomerativo, metodo del legame medio

In questo caso, i clusters ottenuti sono solitamente di piccole dimensioni, ad eccezione di un cluster formato dalla quasi totalità delle osservazioni, a prescindere dal numero di cluster ricercato.

Dal punto di vista delle metriche, i valori dell'indice di Silhouette sono tendenzialmente elevati, vicini a +1, sinonimo di clusters ben definiti e separati gli uni dagli altri; i valori dell'indice di Calinski-Harabasz sono tendenzialmente elevati, sinonimo di clusters separati ed internamente densi, e come per il metodo di Ward tendono a calare all'aumentare del numero di cluster (tuttavia i valori dell'indice sono meno elevati di quelli ottenuti per il metodo di Ward); i valori per l'indice di Davies-Bouldin sono invece molto bassi, tendenti a 0, ad indicare ottima separazione dei clusters; l'indice di Dunn presenta valori vicino allo 0, sinonimo di clustering non ottimale; infine, i valori dell'entropia crescono all'aumentare del numero di cluster.

Ora osserviamo il clustering col metodo del legame completo (*complete linkage*), con uno dei codici eseguiti in *Python* per l'esecuzione dell'algoritmo:

	Silhouette coefficient [-1,1]	Calinski-Harabasz index [0, +∞]	Davies-Bouldin Dunn index [0, +∞]	Entropy index [0, +∞]	Biggest cluster dimension	
2 clusters, dist. euclidea	0.960	1634.266	0.162	0.5932	0.693	49998
3 clusters, dist. Euclidea	0.959	1249.457	0.139	0.1967	1.039	49997
4 clusters, dist. Euclidea	0.929	1263.201	0.393	0.1080	1.951	49992
5 clusters, dist. Euclidea	0.929	1394.337	0.493	0.1255	2.476	49986
10 clusters, dist. Euclidea	0.868	1221.243	0.513	0.1532	3.746	49950
20 clusters, dist. Euclidea	0.772	992.008	0.495	0.2383	10.811	49888
50 clusters, dist. Euclidea	0.320	689.088	0.422	0.0918	10.780	46451
100 clusters, dist. euclidea	0.199	436.768	0.621	0.0820	9.778	45281

```
model_aggcl_2cl_comp_eucl = AgglomerativeClustering(n_clusters=2, linkage='complete',
metric='euclidean')
```

Table 4.14: clustering gerarchico agglomerativo, metodo del legame completo

Anche in questo caso, il comportamento è simile a quanto già visto col metodo del legame medio: i clusters ottenuti sono solitamente di piccole dimensioni, ad eccezione di un cluster formato dalla quasi totalità delle osservazioni, a prescindere dal numero di cluster ricercato. Rispetto al metodo del legame medio, nel caso del legame completo i cluster più piccoli sono comunque formati da un numero più elevato (seppur mediamente sempre molto basso) di istanze.

Per quanto riguarda le metriche considerate (molto simili a quelle ottenute col metodo del legame medio), i valori dell'indice di Silhouette sono tendenzialmente elevati, vicini a +1, sinonimo di clusters ben definiti e separati gli uni dagli altri; i valori dell'indice di Calinski-Harabasz sono tendenzialmente elevati, sinonimo di clusters separati ed internamente densi, e tendono a calare all'aumentare del numero di cluster; anche in questo caso l'indice di Dunn presenta valori vicino allo 0, sinonimo di clustering non ottimale; i valori per l'indice di Davies-Bouldin sono invece molto bassi, tendenti a 0, ad indicare ottima separazione dei clusters; infine, i valori dell'entropia crescono all'aumentare del numero di cluster.

Ora osserviamo il clustering col metodo del legame singolo (*single linkage*), con uno dei codici eseguiti in *Python* per il calcolo del clustering:

	Silhouette coefficient [-1,1]	Calinski-Harabasz index [0, +∞]	Davies-Bouldin Dunn index [0, +∞]	Entropy index [0, +∞]	Biggest cluster dimension	
2 clusters, dist. euclidea	0.960	810.395	0.028	0.5365	0.0	49999
3 clusters, dist. Euclidea	0.959	1249.457	0.139	0.5983	1.055	49997
4 clusters, dist. Euclidea	0.957	1084.623	0.129	0.4854	1.321	49996
5 clusters, dist. Euclidea	0.957	1046.816	0.122	0.4598	1.517	49995
10 clusters, dist. Euclidea	0.926	834.837	0.121	0.4205	2.195	49989
20 clusters, dist. Euclidea	0.898	591.297	0.235	0.2641	2.981	49974
50 clusters, dist. Euclidea	0.748	487.898	0.286	0.2583	4.405	49872
100 clusters, dist. euclidea	0.560	263.610	0.253	0.2389	10.693	49834

```
model_aggcl_2cl_sin_eucl = AgglomerativeClustering(n_clusters=2, linkage='single',
metric='euclidean')
```

Table 4.15: clustering gerarchico agglomerativo, metodo del legame singolo

Anche in questo caso, il comportamento è simile a quanto già visto col metodo del legame medio: i clusters ottenuti sono solitamente di piccole dimensioni, ad eccezione di un cluster formato dalla quasi totalità delle osservazioni, a prescindere dal numero di cluster ricercato, in maniera più simile al clustering col metodo del legame medio rispetto al metodo del legame completo.

Per quanto riguarda le metriche considerate (molto simili a quelle ottenute col metodo del legame medio e del legame completo), i valori dell'indice di Silhouette sono tendenzialmente elevati, vicini a +1, sinonimo di clusters ben definiti e separati gli uni dagli altri; i valori dell'indice di Calinski-Harabasz sono tendenzialmente elevati, sinonimo di clusters separati ed internamente densi, e tendono a calare all'aumentare del numero di cluster; i valori per l'indice di Davies-Bouldin sono invece molto bassi, tendenti a 0, ad indicare ottima separazione dei clusters; anche in questo caso l'indice di Dunn presenta valori vicino allo 0, sinonimo di clustering non ottimale; infine, i valori dell'entropia crescono all'aumentare del numero di cluster.

Infine, osserviamo il clustering gerarchico agglomerativo col metodo del legame singolo (*single linkage*) e distanze diverse da quella euclidea applicata finora, anche in questo caso con uno dei codici eseguiti in *Python* per il calcolo:

	Silhouette coefficient [-1,1]	Calinski-Harabasz index [0, +∞]	Davies-Bouldin Dunn index [0, +∞]	Entropy [0, log2(K)]	Bigest cluster dimension
2 clusters, dist. I2	0.560	0.560	0.253	0.2389	10.693
3 clusters, dist. I2	0.560	0.560	0.253	0.2389	10.693
4 clusters, dist. I2	0.560	0.560	0.253	0.2389	10.693
5 clusters, dist. I2	0.560	0.560	0.253	0.2389	10.693
10 clusters, dist. I2	0.560	0.560	0.253	0.2389	10.693
20 clusters, dist. I2	0.560	0.560	0.253	0.2389	10.693
50 clusters, dist. I2	0.560	0.560	0.253	0.2389	10.693
100 clusters, dist. I2	0.560	0.560	0.253	0.2389	10.693
2 clusters, dist. manhattan	0.527	0.394	0.301	0.1463	10.702
3 clusters, dist. manhattan	0.527	0.394	0.301	0.1463	10.702
4 clusters, dist. manhattan	0.527	0.394	0.301	0.1463	10.702
5 clusters, dist. manhattan	0.527	0.394	0.301	0.1463	10.702
10 clusters, dist. manhattan	0.527	0.394	0.301	0.1463	10.702
20 clusters, dist. manhattan	0.527	0.394	0.301	0.1463	10.702
50 clusters, dist. manhattan	0.527	0.394	0.301	0.1463	10.702
100 clusters, dist. manhattan	0.527	0.394	0.301	0.1463	10.702
2 clusters, dist. coseno	-0.319	-0.357	1.397	0.0106	4.745
3 clusters, dist. coseno	-0.319	-0.357	1.397	0.0106	4.745
4 clusters, dist. coseno	-0.319	-0.357	1.397	0.0106	4.745
5 clusters, dist. coseno	-0.319	-0.357	1.397	0.0106	4.745
10 clusters, dist. coseno	-0.319	-0.357	1.397	0.0106	4.745
20 clusters, dist. coseno	-0.319	-0.357	1.397	0.0106	4.745
50 clusters, dist. coseno	-0.319	-0.357	1.397	0.0106	4.745
100 clusters, dist. coseno	-0.319	-0.357	1.397	0.0106	4.745
model_aggcl_2cl_sin_l1 = AgglomerativeClustering(n_clusters=2, linkage='single', metric='l2')					

Table 4.16: clustering gerarchico agglomerativo, metodo del legame singolo, con metriche diverse dall'euclidea

In tutti i casi appena riportati, a prescindere dalla metrica utilizzata per calcolare la distanza, il comportamento è molto simile ai risultati ottenuti coi metodi del legame medio e del legame singolo: si ottiene in tutti i casi un grande cluster che comprende la quasi totalità delle osservazioni, e una serie di piccolissimi clusters.

Dal punto di vista delle metriche, anche con metriche diverse dalla distanza euclidea i valori dell'indice di Silhouette sono tendenzialmente elevati, vicini a 0.5, sinonimo di clusters definiti e abbastanza separati gli uni dagli altri, a differenza dei risultati ottenuti con la metrica “coseno”, per cui l'indice assume valori negativi pari a -0.319; i valori dell'indice di Calinski-Harabasz sono molto bassi, sinonimo di clusters non ben separati; i valori per l'indice di Davies-Bouldin sono invece bassi, tendenti a 0, ad indicare ottima separazione dei clusters, tranne per i risultati ottenuti con la metrica “coseno”, con valori leggermente più elevati; anche in questo caso finale l'indice di Dunn presenta valori vicino allo 0, sinonimo di clustering non ottimale; infine, i valori dell'entropia sono costantemente intorno al valore +10, ad eccezione di quelli ottenuti con la metrica “coseno”, che si attestano intorno al valore +5.

In base alle metriche osservate, e ai clusters ottenuti, non sembra che il clustering gerarchico agglomerativo riesca a cogliere particolari strutture sottostanti i dati...

4.8.2 BIRCH

Il secondo metodo di cluster analysis applicato è il BIRCH [\[scikit-learn.org\]](https://scikit-learn.org), per cui sono state eseguite una serie di implementazioni con numeri differenti di cluster previsti (2,3,4,5,10,20,50 e 100), e modificando i due fattori *threshold* (raggi dei sottoclusters creati unendo i clusters già esistenti) e *branching_factor* (numero soglia che definisce quando si divide un cluster perché formato da troppe osservazioni); viene anche riportato uno dei codici eseguiti in *Python*, in questo caso per il calcolo del primo risultato con due clusters:

	Silhouette coefficient [-1,1]	Calinski-Harabasz index [0, +∞]	Davies-Bouldin index [0, +∞]	Dunn index [0, +∞]	Entropy [0, log2(K)]	Biggest cluster dim
2 clusters, threshold 0.5, br_fact 50	0.241	6150.417	1.885	0.0059	9.209	40017
3 clusters, threshold 0.5, br_fact 50	0.134	5342.179	2.097	0.0059	10.559	29892
4 clusters, threshold 0.5, br_fact 50	0.133	4368.948	1.612	0.0059	9.852	29892
5 clusters, threshold 0.5, br_fact 50	0.134	3915.061	1.446	0.0064	10.542	29885
10 clusters, threshold 0.5, br_fact 50	0.064	2998.149	1.697	0.0073	10.446	16594
20 clusters, threshold 0.5, br_fact 50	0.073	2621.090	1.093	0.0112	10.103	16594
50 clusters, threshold 0.5, br_fact 50	0.013	1705.317	1.619	0.0213	10.458	6050
100 clusters, threshold 0.5, br_fact 50	-0.003	1094.753	1.889	0.0265	10.511	3713
3 clusters, threshold 0.5, br_fact 50	0.134	5342.179	2.097	0.0059	10.559	29892
3 clusters, threshold 0.1, br_fact 50	0.134	5342.179	2.097	0.0059	10.559	29892
3 clusters, threshold 0.05, br_fact 50	0.105	5630.097	2.184	0.0073	10.531	23444
3 clusters, threshold 0.01, br_fact 50	0.134	5342.179	2.097	0.0059	10.559	29892
3 clusters, threshold 5, br_fact 50	0.929	1441.161	0.452	0.1923	1.9073	49993
3 clusters, threshold 0.5, br_fact 10	0.134	5342.179	2.097	0.0059	10.559	29892
3 clusters, threshold 0.5, br_fact 100	0.134	5342.179	2.097	0.0059	10.559	29892
3 clusters, threshold 0.5, br_fact 5	0.134	5342.179	2.097	0.0059	10.559	29892
3 clusters, threshold 0.5, br_fact 500	0.134	5342.179	2.097	0.0059	10.559	29892

```
model_birch_2cl_05thr_50bra = Birch(threshold=0.5, branching_factor=50, n_clusters=2)
```

Table 4.17: BIRCH clustering

È possibile notare che i risultati ottenuti dal clustering non sono cambiati molto (mantenendo costante il numero di cluster) al variare degli altri parametri (ad eccezione di valori particolarmente elevati per il parametro *threshold*, come nel caso di *threshold* = 5).

In generale, in caso di pochi clusters richiesti come risultato finale, si andavano ad ottenere gruppi composti da un numero significativo di osservazioni (in quasi tutti i casi compresi tra 5000 e 30000 istanze).

All'aumentare del numero di cluster ricercati, andavano a crearsi una serie di clusters composti da pochissime osservazioni.

Per quanto riguarda le metriche considerate, i valori dell'indice di Silhouette sono tendenzialmente bassi (tranne in un caso), sinonimo di clusters non ben definiti e separati gli uni dagli altri; i valori dell'indice di Calinski-Harabasz, invece, sono tendenzialmente elevati, sinonimo di clusters separati ed internamente densi, e tendono a calare all'aumentare del numero di cluster; i valori per l'indice di Davies-Bouldin sono invece più elevati di quelli visti nel clustering gerarchico agglomerativo, ad indicare solamente una buona separazione dei clusters; anche in questo caso finale l'indice di Dunn presenta valori vicino allo 0, sinonimo di clustering non ottimale; infine, i valori dell'entropia si mantengono costanti anche all'aumentare del numero di cluster.

Anche il BIRCH, come il clustering agglomerativo gerarchico, non sembra riesca a rappresentare ed individuare particolari strutture sottostanti i dati...

4.8.3 DBSCAN

Il terzo metodo applicato è il DBSCAN [\[scikit-learn.org\]](https://scikit-learn.org), per cui sono state eseguite una serie di implementazioni modificando i due fattori *eps* (massima distanza tra due punti per essere considerati vicini) e *min_samples* (numero di punti in uno spazio del piano perché un punto possa essere considerato “core point”); viene anche riportato uno dei codici eseguiti in *Python*, in questo caso per il calcolo del primo risultato riportato nella seguente tabella:

	n° cluster	Silhouette coefficient [-1,1]	Calinski-Harabasz index [0, +∞]	Davies-Bouldin index [0, +∞]	Dunn index [0, +∞]	Entropy [0, log2(K)]	Bigest cluster dimension
Eps 100, min_s 5	2	0.960	1356.028	0.890	0.3998	1.099	49997
Eps 50, min_s 5	2	0.938	1067.796	2.628	0.1985	3.135	49977
Eps 150, min_s 5	2	0.960	810.396	0.028	0.0829	0.0	49999
Eps 100, min_s 10	2	0.960	1356.028	0.891	0.1254	1.099	49997
Eps 100, min_s 3	2	0.960	1356.028	0.891	0.1524	1.099	49997
Eps 100, min_s 20	2	0.959	1394.254	1.378	0.1942	1.609	49995
Eps 100, min_s 50	2	0.959	1394.254	1.378	0.1026	1.609	49995

```
model_dbSCAN_eps100_5mins = DBSCAN(eps=100, min_samples=5, metric='euclidean')
```

Table 4.18: DBSCAN clustering

Tutti i risultati ottenuti dalle diverse implementazioni del DBSCAN hanno formato un cluster composto dalla maggioranza delle osservazioni, e una seconda classe (definita come rumore, o “noise”) indicata come -1, e formata dalle rimanenti osservazioni.

Per quanto riguarda le metriche considerate, i valori dell'indice di Silhouette sono tendenzialmente elevati (vicini a +1), sinonimo di clusters ben definiti e ben separati gli uni dagli altri; i valori dell'indice di Calinski-Harabasz, invece, sono tendenzialmente elevati, sinonimo di clusters separati ed internamente densi, e si mantengono costanti al cambiare i parametri dell'algoritmo; i valori per l'indice di Davies-

Bouldin assumono valori in linea con quelli visti nel clustering gerarchico agglomerativo, ad indicare solamente un'ottima separazione dei clusters; anche in questo caso finale l'indice di Dunn presenta valori vicino allo 0, sinonimo di clustering non ottimale; infine, i valori dell'entropia si mantengono costanti, e assumono valori abbastanza vicini allo 0.

Il metodo DBSCAN è forse il metodo peggiore di clustering visto finora, infatti non riesce a catturare la struttura di nessun cluster al di fuori di quello “totale”: l'unica utilità possibile dell'algoritmo, visti i risultati ottenuti, sarebbe per individuare ed escludere le osservazioni definite come “noise”, che potrebbero inficiare i risultati di tutti gli algoritmi di cluster analysis e riduzione della dimensionalità.

4.8.4 K-means

Infine, osserviamo il metodo K-means (o delle k-medie) [\[scikit-learn.org\]](http://scikit-learn.org) con una serie di implementazioni modificando i quattro fattori *n_cluster* (per il numero di cluster, scelti pari a 2,3,4 e 5), *n_init* (numero di volte che l'algoritmo è rieseguito con centroidi diversi) *max_iter* (numero massimo di iterazioni ogni volta che l'algoritmo viene eseguito) e *tol* (tolleranza relativa riguardo la norma di Frobenius"); viene anche riportato uno dei codici eseguiti in *Python*, in questo caso per il calcolo del primo risultato riportato nella seguente tabella:

	Silhouette coefficient [-1,1]	Calinski-Harabasz index [0, +∞]	Davies-Bouldin index [0, +∞]	Dunn index [0, +∞]	Entropy [0, log2(K)]	Biggest cluster dimension
2 clusters, n_init 100, m_iter 1000, tol 0.01	0.202	9179.551	1.759	0.0064	9.991	28238
3 clusters, n_init 100, m_iter 1000, tol 0.01	0.150	6648.574	1.949	0.0045	10.540	25616
4 clusters, n_init 100, m_iter 1000, tol 0.01	0.151	5309.132	1.499	0.0048	9.960	25483
5 clusters, n_init 100, m_iter 1000, tol 0.01	0.119	4589.101	1.804	0.0045	10.121	20857
2 clusters, n_init 100, m_iter 100, tol 0.01	0.203	9178.979	1.758	0.0061	10.255	28423
3 clusters, n_init 100, m_iter 100, tol 0.01	0.149	6648.456	1.956	0.0048	10.439	25394
4 clusters, n_init 100, m_iter 100, tol 0.01	0.149	5312.395	1.506	0.0048	10.052	25370
5 clusters, n_init 100, m_iter 100, tol 0.01	0.149	4678.013	1.348	0.0045	10.090	25392
2 clusters, n_init 100, m_iter 10, tol 0.01	0.203	9179.473	1.759	0.0064	9.986	28271
3 clusters, n_init 100, m_iter 10, tol 0.01	0.149	6648.547	1.955	0.0045	10.418	25404
4 clusters, n_init 100, m_iter 10, tol 0.01	0.148	5311.226	1.506	0.0048	10.405	25428
5 clusters, n_init 100, m_iter 10, tol 0.01	0.112	4588.403	1.818	0.0050	10.593	19983
2 clusters, n_init 100, m_iter 10000, tol 0.01	0.202	9179.483	1.759	0.0064	9.988	28242
3 clusters, n_init 100, m_iter 10000, tol 0.01	0.149	6648.585	1.954	0.0045	10.437	25448
4 clusters, n_init 100, m_iter 10000, tol 0.01	0.149	5312.512	1.501	0.0048	10.542	25573
5 clusters, n_init 100, m_iter 10000, tol 0.01	0.114	4589.079	1.813	0.0045	10.447	20287
2 clusters, n_init 100, m_iter 1000, tol 0.1	0.201	9178.586	1.759	0.0061	10.235	27867
3 clusters, n_init 100, m_iter 1000, tol 0.1	0.149	6648.159	1.955	0.0045	10.548	25433
4 clusters, n_init 100, m_iter 1000, tol 0.1	0.147	5307.704	1.508	0.0048	10.460	25321
5 clusters, n_init 100, m_iter 1000, tol 0.1	0.123	4576.816	1.812	0.0048	10.422	20843
2 clusters, n_init 100, m_iter 1000, tol 1	0.202	9179.452	1.759	0.0064	10.247	28191
3 clusters, n_init 100, m_iter 1000, tol 1	0.151	6647.325	1.945	0.0047	10.419	25759
4 clusters, n_init 100, m_iter 1000, tol 1	0.148	5306.823	1.530	0.0048	10.073	25097
5 clusters, n_init 100, m_iter 1000, tol 1	0.146	4689.240	1.271	0.0048	10.497	25131
2 clusters, n_init 100, m_iter 1000, tol 0.001	0.202	9179.549	1.759	0.0064	9.990	28191
3 clusters, n_init 100, m_iter 1000, tol 0.001	0.149	6648.593	1.953	0.0045	10.057	25482
4 clusters, n_init 100, m_iter 1000, tol 0.001	0.149	5312.616	1.502	0.0048	10.085	25531
5 clusters, n_init 100, m_iter 1000, tol 0.001	0.149	4698.475	1.265	0.0050	10.496	25444
2 clusters, n_init 1000, m_iter 1000, tol 0.01	0.204	9177.079	1.758	0.0065	9.965	28162
3 clusters, n_init 1000, m_iter 1000, tol 0.01	0.149	6647.768	1.955	0.0045	10.057	25426

4 clusters, n_init 1000, m_iter 1000, tol 0.01	0.150	5309.434	1.499	0.0048	9.960	25430
5 clusters, n_init 1000, m_iter 1000, tol 0.01	0.111	4586.359	1.816	0.0054	10.496	25483
2 clusters, n_init 10, m_iter 1000, tol 0.01	0.202	0.202	1.759	0.0061	10.246	28734
3 clusters, n_init 10, m_iter 1000, tol 0.01	0.149	0.149	1.955	0.0048	10.435	25439
4 clusters, n_init 10, m_iter 1000, tol 0.01	0.149	0.149	1.502	0.0048	10.425	25490
5 clusters, n_init 10, m_iter 1000, tol 0.01	0.149	0.149	1.264	0.0048	9.962	20065

```
model_kmeans_2cl_100in_1000maxi_001tol = KMeans(n_clusters=2, n_init = 100,
algorithm='lloyd', max_iter = 1000, tol = 0.01)
```

Table 4.19: K-means clustering

Rispetto agli altri metodi, si tratta dell'algoritmo che (assieme al clustering gerarchico agglomerativo col metodo di Ward, e al BIRCH) crea clusters di dimensioni più consistenti tra di loro, soprattutto se si ricercano 2 o 3 clusters come risultato finale. Invece, quando la scelta ricade su 4 o 5 clusters (o anche per numeri superiori, come testato in questi mesi e non riportato in questo elaborato per la poca utilità e qualità dei risultati ottenuti), vengono formati un 4° ed un 5° cluster di piccolissime dimensioni (solitamente nell'ordine dell'unità).

Per quanto riguarda le metriche considerate, i valori dell'indice di Silhouette sono tendenzialmente bassi (vicini allo 0), sinonimo di clusters non ben definiti e non ben separati gli uni dagli altri; i valori dell'indice di Calinski-Harabasz, invece, sono tendenzialmente elevati, sinonimo di clusters separati ed internamente densi, tranne quando il parametro *n_init* viene fissato con valori particolarmente bassi; i valori per l'indice di Davies-Bouldin sono invece più elevati di quelli visti nel clustering gerarchico agglomerativo, ad indicare solamente una buona separazione dei clusters; anche in questo caso finale l'indice di Dunn presenta valori vicino allo 0, sinonimo di clustering non ottimale; infine, i valori dell'entropia si mantengono costanti, e assumono valori abbastanza vicini a 10.

Si tratta dell'algoritmo che riesce meglio a ricreare clusters di dimensioni significative, soprattutto se confrontato con gli altri metodi. Inoltre, è particolarmente efficiente dal punto di vista computazionale, con algoritmi eseguiti in pochi secondi sul campione da 50000 osservazioni.

4.8.5 Spectral Clustering

Sono stati fatti dei tentativi per implementare il metodo dello Spectral Clustering [[scikit-learn.org](#)]: tuttavia, si tratta di un algoritmo che, dal punto di vista computazionale, è risultato molto più “dispendioso” a livello di tempistiche rispetto ad altri metodi (andando a costruire uno spazio di dimensioni superiori a quello in input, è normale osservare problemi simili con un dataset delle dimensioni di quello in studio).

Infatti, anche solo per un campione formato da 5000 osservazioni (invece delle 50000 utilizzate per tutti gli altri metodi), l'algoritmo ha richiesto circa due ore e mezzo per completare il clustering, a fronte di un massimo di 25 minuti circa per tutti gli altri metodi utilizzati col campione di 50000. Dunque, non sono stati riportati i risultati ottenuti sul sottocampione da 5000 osservazioni, in quanto non direttamente confrontabili con gli altri metodi applicati.

5. Risultati e discussione cluster analysis dopo riduzione di dimensionalità

Con la fase di riduzione della dimensionalità l’obiettivo era creare un nuovo dataset di input per i dati che mantenesse comunque la capacità comunicativa e la quantità di varianza presente nei dati; con la cluster analysis, si puntava a creare clusters densi internamente e ben separati dagli altri esternamente, in modo da riconoscere gruppi di osservazioni con caratteristiche chiare e definite.

Le due metodologie di algoritmi di machine learning possono lavorare in combinazione per arrivare a risultati significativi, sfruttando a vicenda le caratteristiche e i punti di forza dell’altra: infatti le informazioni ottenute dalle fasi precedenti del progetto possono essere utili ora per indirizzare le scelte su metodi veloci, efficaci, computazionalmente fattibili per i dispositivi a disposizione ed altamente informativi.

Inoltre, effettuando riduzione di dimensionalità sui dati originali, è possibile ora effettuare cluster analysis sull’intero campione a disposizione, non limitandosi più al solo campione di 50000 osservazioni come nel capitolo [4.8].

5.1.1 UMAP

Viene effettuata cluster analysis col metodo delle k medie, con 2/3/4/5 clusters, applicata ai dati completi su cui è stata effettuata riduzione di dimensionalità col metodo *UMAP*($n_neighbors = 15$, $min_dist = 0.1$).

Nel primo caso, il cluster 0 ha numerosità 213931, il cluster 1 è composto da 58183 osservazioni.

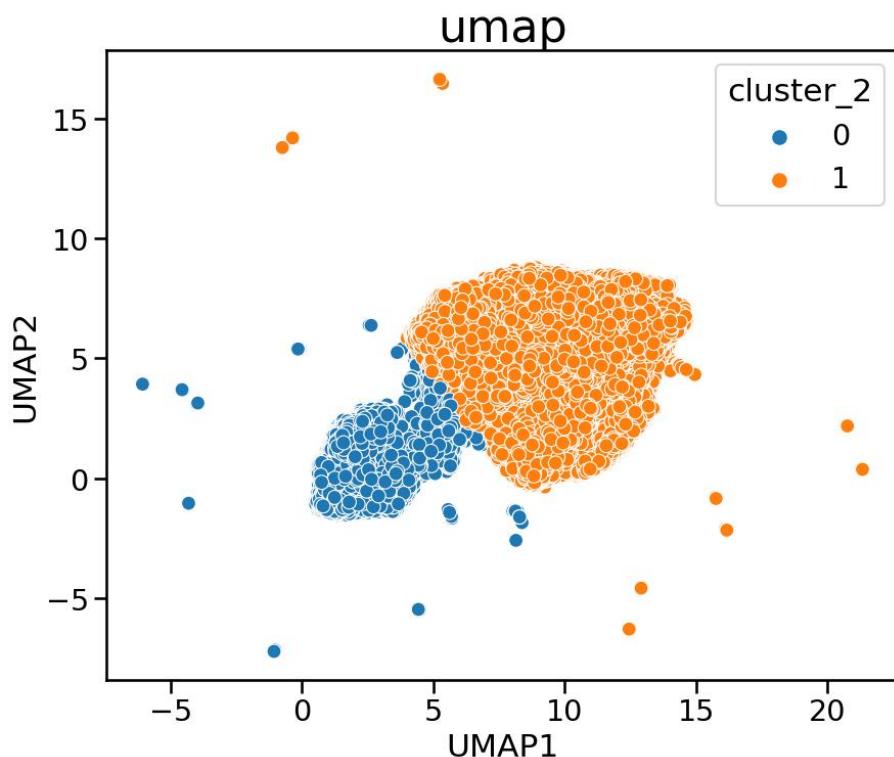


Figure 5.21: scatterplot K-means (2 clusters) con dati ridotti UMAP

Il clustering sembra dividere in due parti la regione densa, assegnando poi i valori lontani dalla “nuvola” di dati in parte ad uno ed in parte all’altro cluster.

	Silhouette coefficient [-1,1]	Calinski-Harabasz index [0, +∞]	Davies-Bouldin index [0, +∞]	Dunn index [0, +∞]	Entropy [0, log2(K)]
K-means(2 clusters)	0.543	331062.561	0.571	0.241	12.273

Table 5.20: indici K-means (2 clusters) con dati ridotti UMAP

Per quanto riguarda le metriche considerate, il valore dell’indice di Silhouette è piuttosto basso (vicino allo 0.5), sinonimo di clusters abbastanza ben definiti e abbastanza separati gli uni dagli altri; l’indice di Calinski-Harabasz, invece, assume valore molto elevato, sinonimo di clusters ben separati ed internamente molto densi; il valore dell’indice di Davies-Bouldin invece è basso, ad indicare un’ottima separazione dei clusters; il valore dell’indice di Dunn è basso come in tutte le casistiche precedenti; infine, l’entropia assume valore pari a 12.273.

Nel secondo caso, il cluster 0 è formato da 109282 osservazioni, il cluster 1 da 57609 e il cluster 2 da 105223.

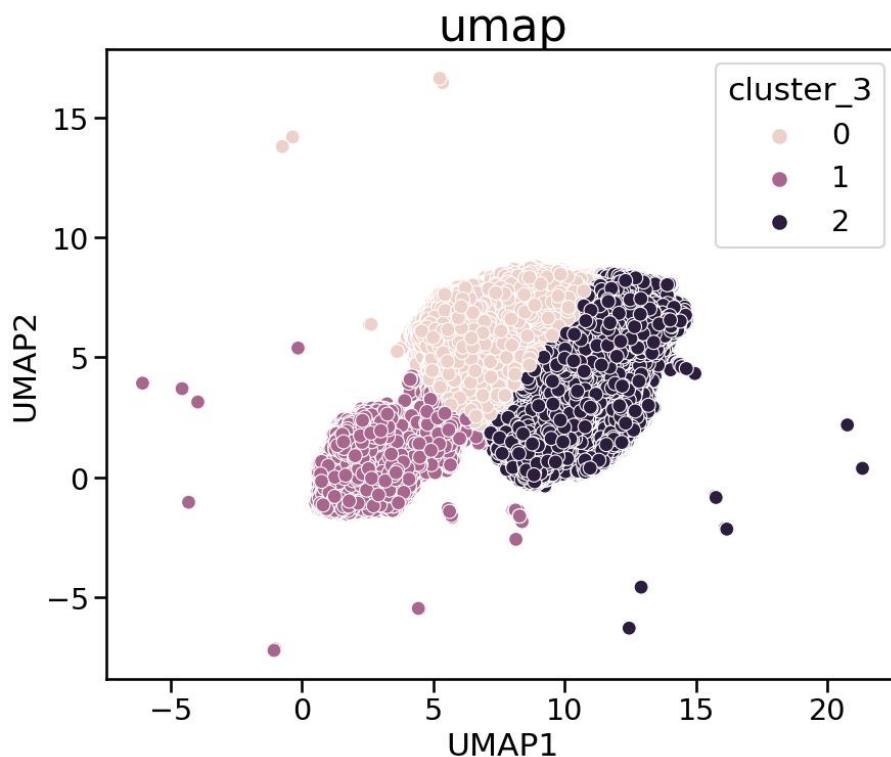


Figure 5.22: scatterplot K-means (3 clusters) con dati ridotti UMAP

Come nel primo caso, il clustering sembra dividere in tre parti la densa regione centrale, assegnando poi i valori lontani dalla “nuvola” di dati ai vari clusters ottenuti.

	Silhouette coefficient [-1,1]	Calinski-Harabasz index [0, +∞]	Davies-Bouldin index [0, +∞]	Dunn index [0, +∞]	Entropy [0, log2(K)]
--	-------------------------------	---------------------------------	------------------------------	--------------------	----------------------

K-means(3 clusters)	0.449	350882.159	0.879	0.311	11.954
---------------------	-------	------------	-------	-------	--------

Table 5.21: indici K-means (3 clusters) con dati ridotti UMAP

Per quanto riguarda le metriche considerate, il valore dell’indice di Silhouette è piuttosto basso (vicino allo 0.5), sinonimo di clusters abbastanza ben definiti e abbastanza separati gli uni dagli altri; l’indice di Calinski-Harabasz, invece, assume valore molto elevato (anche più dei K-means con due clusters), sinonimo di clusters ben separati ed internamente molto densi; il valore dell’indice di Davies-Bouldin invece è basso, ad indicare un’ottima separazione dei clusters; il valore dell’indice di Dunn è basso come in tutte le casistiche precedenti; infine, l’entropia assume valore pari a 11.954.

Nel terzo caso, il cluster 0 è formato da 83436 osservazioni, il cluster 1 da 56637, il cluster 2 da 79371 e il cluster 3 da 52670 osservazioni.

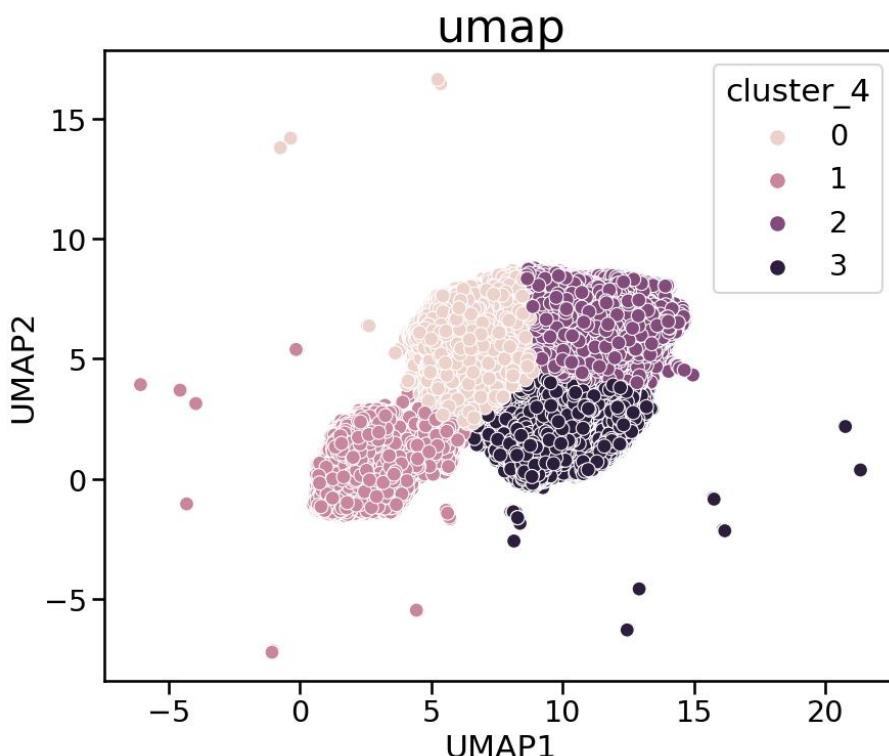


Figure 5.23: scatterplot K-means (4 clusters) con dati ridotti UMAP

Come nei casi precedenti, il clustering sembra dividere in quattro parti abbastanza bilanciate la densa regione centrale, assegnando poi i valori lontani dalla “nuvola” ai vari clusters ottenuti.

	Silhouette coefficient [-1,1]	Calinski-Harabasz index [0, +∞]	Davies-Bouldin index [0, +∞]	Dunn index [0, +∞]	Entropy [0, log2(K)]
K-means(4 clusters)	0.512	471294.575	0.674	0.271	12.070

Table 5.22: indici K-means (4 clusters) con dati ridotti UMAP

Per quanto riguarda le metriche considerate, il valore dell’indice di Silhouette è piuttosto basso (vicino allo 0.5), sinonimo di clusters abbastanza ben definiti e abbastanza separati gli uni dagli altri; l’indice di Calinski-Harabasz, invece, assume valore molto elevato (anche più dei K-means precedenti), sinonimo di

clusters ben separati ed internamente molto densi; il valore dell’indice di Davies-Bouldin invece è basso, ad indicare un’ottima separazione dei clusters; il valore dell’indice di Dunn è basso come in tutte le casistiche precedenti; infine, l’entropia assume valore pari a 12.070.

Infine, nel quarto caso, il cluster 0 è formato da 59009 osservazioni, il cluster 1 da 56044, il cluster 2 da 51266, il cluster 3 da 59282 e il cluster 4 da 46513 osservazioni.

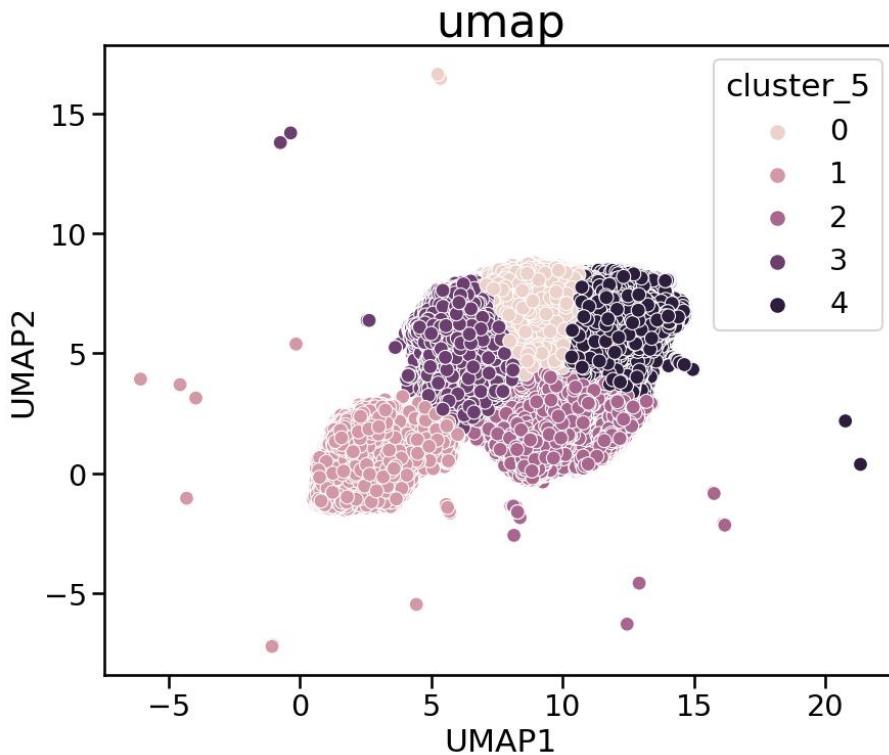


Figure 5.24: scatterplot K-means (5 clusters) con dati ridotti UMAP

Anche in quest’ultimo caso, il clustering sembra dividere in cinque parti molto bilanciate la densa regione centrale, assegnando poi i valori lontani dalla “nuvola” ai vari clusters ottenuti.

	Silhouette coefficient [-1,1]	Calinski-Harabasz index [0, +∞]	Davies-Bouldin index [0, +∞]	Dunn index [0, +∞]	Entropy [0, log2(K)]
K-means(5 clusters)	0.471	457845.941	0.751	0.195	12.162

Table 5.23: indici K-means (5 clusters) con dati ridotti UMAP

Per quanto riguarda le metriche considerate, il valore dell’indice di Silhouette è piuttosto basso (vicino allo 0.5), sinonimo di clusters abbastanza ben definiti e abbastanza separati gli uni dagli altri; l’indice di Calinski-Harabasz, invece, assume valore molto elevato (simile ai casi precedenti), sinonimo di clusters ben separati ed internamente molto densi; il valore dell’indice di Davies-Bouldin invece è basso, ad indicare un’ottima separazione dei clusters; il valore dell’indice di Dunn è basso come in tutte le casistiche precedenti; infine, l’entropia assume valore pari a 12.162.

5.1.2 PCA

Viene effettuata cluster analysis col metodo delle k medie, con 2/3/4/5 clusters, applicata ai dati completi su cui è stata effettuata riduzione di dimensionalità col metodo *PCA()*.

Nel primo caso, il cluster 0 ha numerosità 214946, il cluster 1 è composto da 70768 osservazioni.

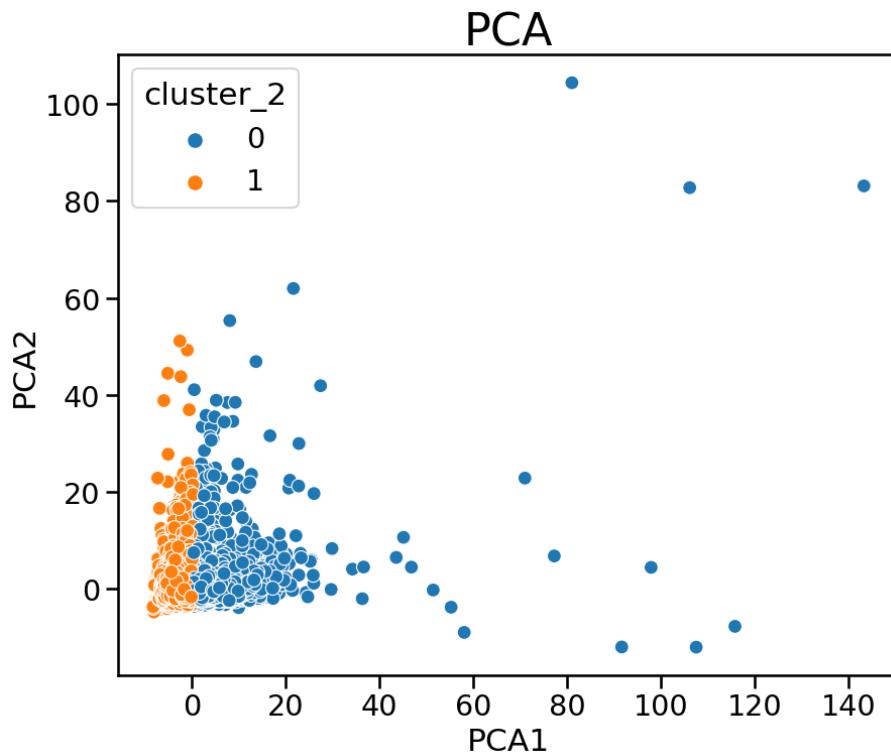


Figure 5.25: scatterplot K-means (2 clusters) con dati ridotti PCA

Con la PCA e due clusters, il clustering sembra suddividere in due parti la regione molto fitta di osservazioni, assegnando poi i valori lontani dalla “nuvola” ai due clusters ottenuti.

	Silhouette coefficient [-1,1]	Calinski-Harabasz index [0, +∞]	Davies-Bouldin index [0, +∞]	Dunn index [0, +∞]	Entropy [0, log2(K)]
K-means(2 clusters)	0.431	252476.199	0.853	0.263	12.005

Table 5.24: indici K-means (2 clusters) con dati ridotti PCA

Per quanto riguarda le metriche considerate, il valore dell’indice di Silhouette è piuttosto basso (vicino allo 0.5), sinonimo di clusters abbastanza ben definiti e abbastanza separati gli uni dagli altri; l’indice di Calinski-Harabasz, invece, assume valore molto elevato, sinonimo di clusters ben separati ed internamente molto densi; il valore dell’indice di Davies-Bouldin invece è basso, ad indicare un’ottima separazione dei clusters; il valore dell’indice di Dunn è basso come in tutte le casistiche precedenti; infine, l’entropia assume valore pari a 12.231. I valori sono tendenzialmente in linea con quanto visto nel caso dell’UMAP come metodo di riduzione della dimensionalità.

Nel secondo caso, il cluster 0 è formato da 91489 osservazioni, il cluster 1 da 135623 e il cluster 2 da 58602.

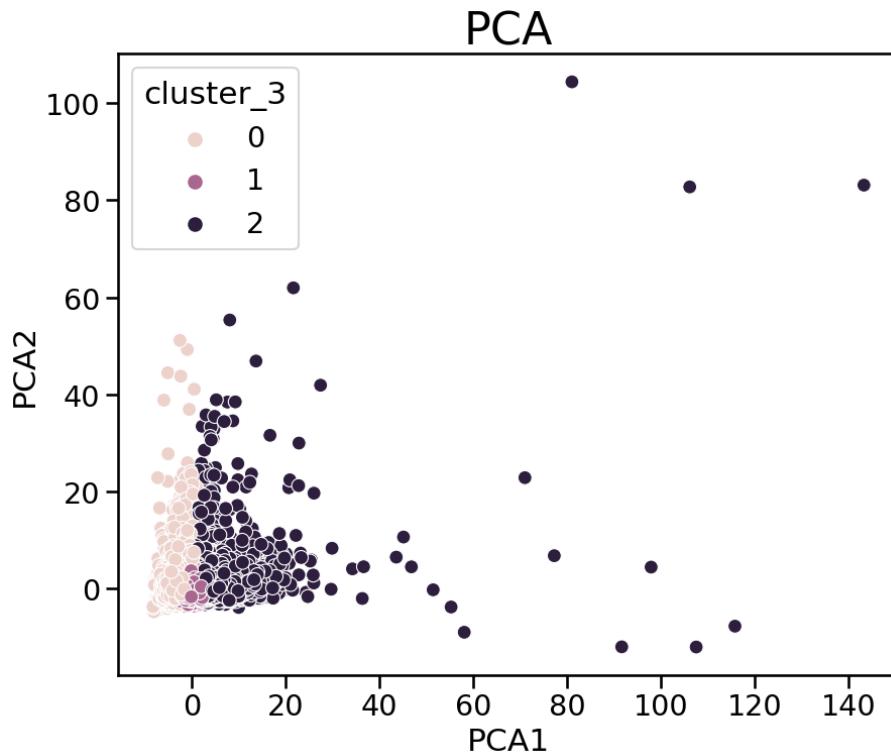


Figure 5.26: scatterplot K-means (3 clusters) con dati ridotti PCA

Nel secondo caso della PCA, il clustering sembra suddividere in tre parti la regione molto fitta di osservazioni (creando un piccolo cluster per il gruppo 1 molto denso e fitto), assegnando poi i valori lontani dalla “nuvola” ai clusters rimanenti.

	Silhouette coefficient [-1,1]	Calinski-Harabasz index [0, +∞]	Davies-Bouldin index [0, +∞]	Dunn index [0, +∞]	Entropy [0, log2(K)]
K-means(3 clusters)	0.393	232498.274	0.866	0.278	12.119

Table 5.25: indici K-means (3 clusters) con dati ridotti PCA

Per quanto riguarda le metriche considerate, il valore dell’indice di Silhouette è piuttosto basso (vicino allo 0.5), sinonimo di clusters abbastanza ben definiti e abbastanza separati gli uni dagli altri; l’indice di Calinski-Harabasz, invece, assume valore molto elevato (anche più del K-means con due clusters), sinonimo di clusters ben separati ed internamente molto densi; il valore dell’indice di Davies-Bouldin invece è basso, ad indicare un’ottima separazione dei clusters; il valore dell’indice di Dunn è tendenzialmente basso, come in tutte le casistiche precedenti; infine, l’entropia assume valore pari a 12.231. Anche in questo caso i valori sono tendenzialmente in linea con quanto visto nel caso dell’UMAP come metodo di riduzione della dimensionalità.

Osserviamo ora il terzo caso, con il cluster 0 formato da 89971 osservazioni, il cluster 1 da 71366, il cluster 2 da 92440, e il cluster 3 formato da 31937 osservazioni.

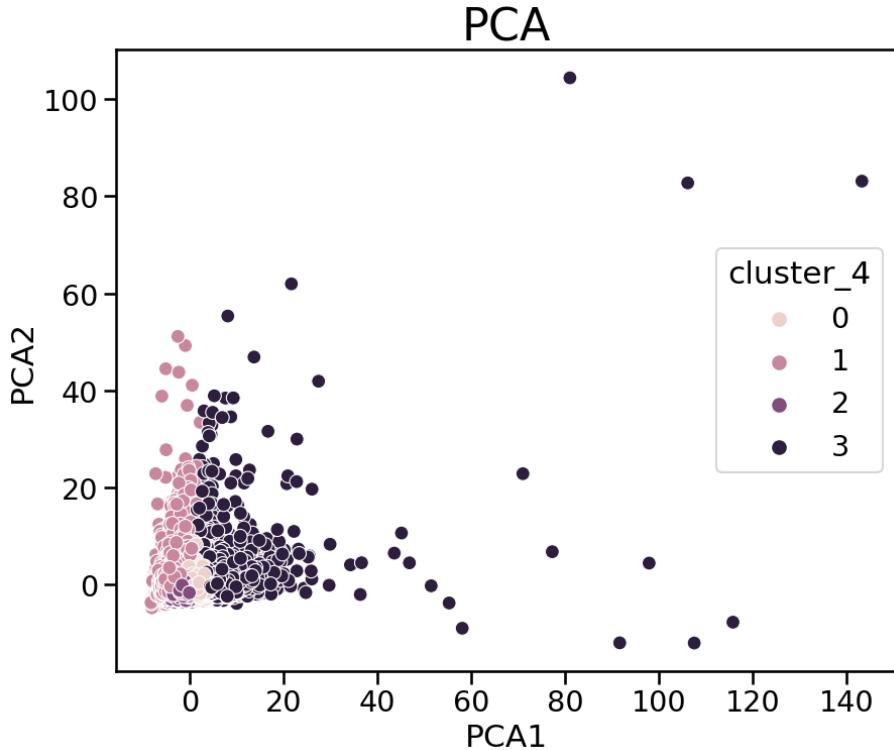


Figure 5.27: scatterplot K-means (4 clusters) con dati ridotti PCA

Nel terzo caso della PCA, il clustering sembra suddividere in quattro parti la regione molto fitta di osservazioni (in alcuni casi con sottoregioni molto ristrette, come per il cluster 0 e 2), assegnando poi i valori lontani dalla “nuvola” principalmente a due soli clusters.

	Silhouette coefficient [-1,1]	Calinski-Harabasz index [0, +∞]	Davies-Bouldin index [0, +∞]	Dunn index [0, +∞]	Entropy [0, log2(K)]
K-means(4 clusters)	0.344	206574.963	0.922	0.187	12.108

Table 5.26: indici K-means (4 clusters) con dati ridotti PCA

Per quanto riguarda le metriche considerate, il valore dell’indice di Silhouette è piuttosto basso (vicino allo 0.5), sinonimo di clusters abbastanza ben definiti e abbastanza separati gli uni dagli altri; l’indice di Calinski-Harabasz, invece, assume valore molto elevato (in linea con i valori visti col K-means con due e tre clusters), sinonimo di clusters ben separati ed internamente molto densi; il valore dell’indice di Davies-Bouldin invece è basso, ad indicare un’ottima separazione dei clusters; il valore dell’indice di Dunn è tendenzialmente basso, come in tutte le casistiche precedenti; infine, l’entropia assume valore pari a 12.108. Anche in questo caso i valori sono tendenzialmente in linea con quanto visto nel caso dell’UMAP come metodo di riduzione della dimensionalità.

Osserviamo ora il quarto caso e ultimo caso della PCA, con il cluster 0 formato da 88221 osservazioni, il cluster 1 da 60922, il cluster 2 da 31728, il cluster 3 formato da 14484 osservazioni e il cluster 4 formato da 90359 osservazioni.

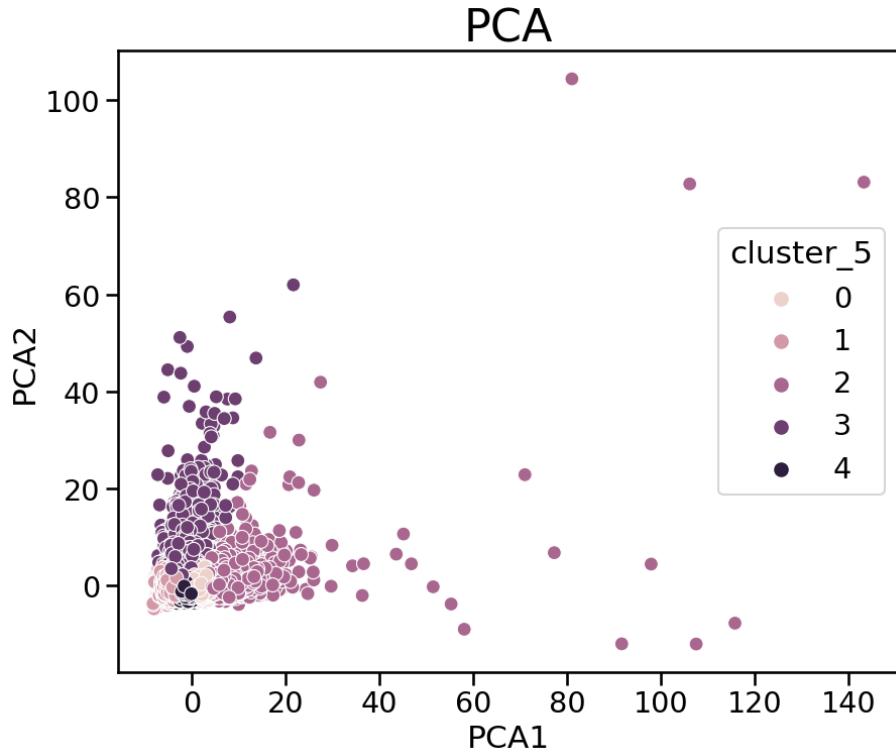


Figure 5.28: scatterplot K-means (5 clusters) con dati ridotti PCA

Nell'ultimo caso della PCA, il clustering sembra comportarsi come nel caso precedente, con la regione molto fitta di osservazioni divisa in cinque parti (osservazioni molto compatte per i clusters 0, 1 e 4), assegnando poi i valori lontani dalla “nuvola” principalmente a due soli clusters.

	Silhouette coefficient [-1, 1]	Calinski-Harabasz index [0, +∞]	Davies-Bouldin index [0, +∞]	Dunn index [0, +∞]	Entropy [0, log2(K)]
K-means(5 clusters)	0.365	206816.010	0.855	0.175	12.059

Table 5.27: indici K-means (5 clusters) con dati ridotti PCA

Per quanto riguarda le metriche considerate, il valore dell'indice di Silhouette è piuttosto basso (vicino allo 0.5), sinonimo di clusters abbastanza ben definiti e abbastanza separati gli uni dagli altri; l'indice di Calinski-Harabasz, invece, assume valore molto elevato (in linea con i valori visti in precedenza), sinonimo di clusters ben separati ed internamente molto densi; il valore dell'indice di Davies-Bouldin invece è basso, ad indicare un'ottima separazione dei clusters; il valore dell'indice di Dunn è tendenzialmente basso, come in tutte le casistiche precedenti; infine, l'entropia assume valore pari a 12.059. Anche in questo caso i valori sono tendenzialmente in linea con quanto visto nel caso dell'UMAP come metodo di riduzione della dimensionalità.

5.1.3 t-SNE

In questa terza fase e ultima fase del clustering applicato a dati ridotti, viene effettuata cluster analysis col metodo delle k medie, con 2/3/4/5 clusters, ai dati completi su cui è stata effettuata riduzione di dimensionalità col metodo *T-sne*(*perplexity* = 50, *early_exaggeration* = 30).

Nel primo caso, il cluster 0 ha numerosità 137500, il cluster 1 è composto da 148214 osservazioni.

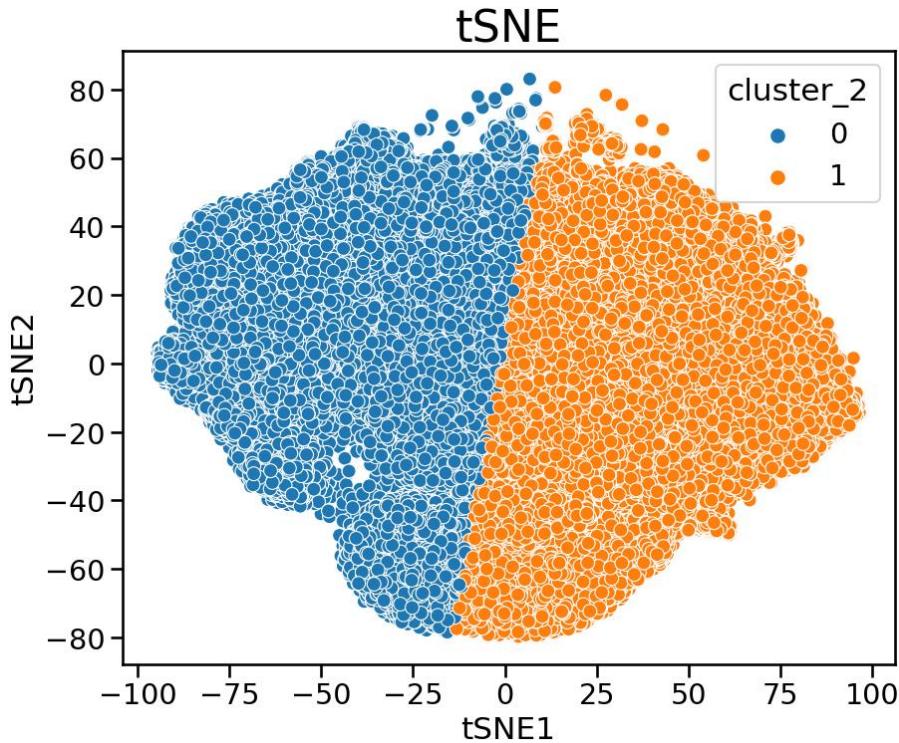


Figure 5.29: scatterplot K-means (2 clusters) con dati ridotti t-SNE

Osserviamo che l'algoritmo di clustering sembra dividere la regione molto fitta di osservazioni in due parti, assegnando i pochissimi valori lontani dalla “nuvola” principale al cluster più vicino.

	Silhouette coefficient [-1,1]	Calinski-Harabasz index [0, +∞]	Davies-Bouldin index [0, +∞]	Dunn index [0, +∞]	Entropy [0, log2(K)]
K-means(2 clusters)	0.391	235425.807	1.007	0.159	11.906

Table 5.28: indici K-means (2 clusters) con dati ridotti t-SNE

Per quanto riguarda le metriche considerate, il valore dell'indice di Silhouette è piuttosto basso (vicino allo 0.5), sinonimo di clusters abbastanza ben definiti e abbastanza separati gli uni dagli altri; l'indice di Calinski-Harabasz, invece, assume valore molto elevato (anche più del K-means con due clusters), sinonimo di clusters ben separati ed internamente molto densi; il valore dell'indice di Davies-Bouldin invece è basso, ad indicare un'ottima separazione dei clusters; il valore dell'indice di Dunn è tendenzialmente basso, come in tutte le casistiche precedenti; infine, l'entropia assume valore pari a 11.906. Anche in questo caso i valori sono tendenzialmente in linea con quanto visto nel caso dell'UMAP e della PCA come metodi di riduzione della dimensionalità.

Nel secondo caso invece, il cluster 0 ha numerosità 70881, il cluster 1 è composto da 108283 osservazioni e il cluster 2 è composto da 106550 osservazioni.

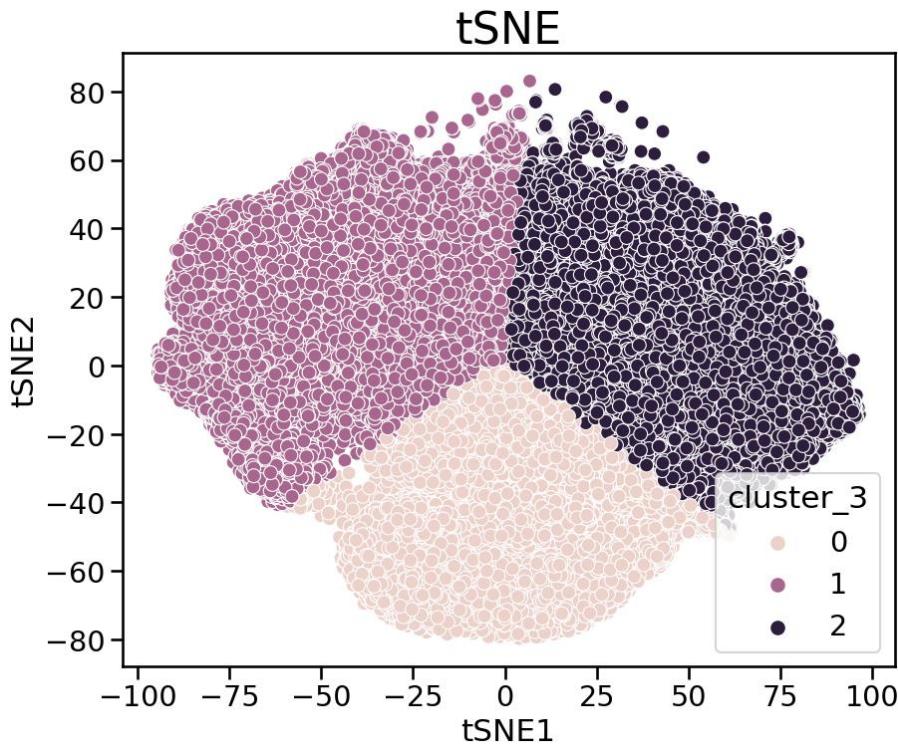


Figure 5.30: scatterplot K-means (3 clusters) con dati ridotti t-SNE

È possibile osservare che la divisione effettuata dal clustering porta alla regione molto fitta di osservazioni divisa in tre parti, assegnando i pochissimi valori lontani dalla “nuvola” principale al cluster più vicino.

	Silhouette coefficient [-1,1]	Calinski-Harabasz index [0, +∞]	Davies-Bouldin index [0, +∞]	Dunn index [0, +∞]	Entropy [0, log2(K)]
K-means(3 clusters)	0.404	266677.214	0.843	0.229	12.221

Table 5.29: indici K-means (3 clusters) con dati ridotti t-SNE

Per quanto riguarda le metriche considerate, il valore dell’indice di Silhouette è piuttosto basso (vicino allo 0.5), sinonimo di clusters abbastanza ben definiti e abbastanza separati gli uni dagli altri; l’indice di Calinski-Harabasz, invece, assume valore molto elevato (anche più del K-means con due clusters), sinonimo di clusters ben separati ed internamente molto densi; il valore dell’indice di Davies-Bouldin invece è basso, ad indicare un’ottima separazione dei clusters; il valore dell’indice di Dunn è tendenzialmente basso, come in tutti i casi precedenti; infine, l’entropia assume valore pari a 12.221. Anche in questo caso i valori sono tendenzialmente in linea con quanto visto nel caso dell’UMAP e della PCA come metodi di riduzione della dimensionalità.

Nel terzo caso invece, il cluster 0 ha numerosità 76934, il cluster 1 è composto da 68687 osservazioni, il cluster 2 è composto da 70285 osservazioni e il cluster 3 è composto da 69808 osservazioni.

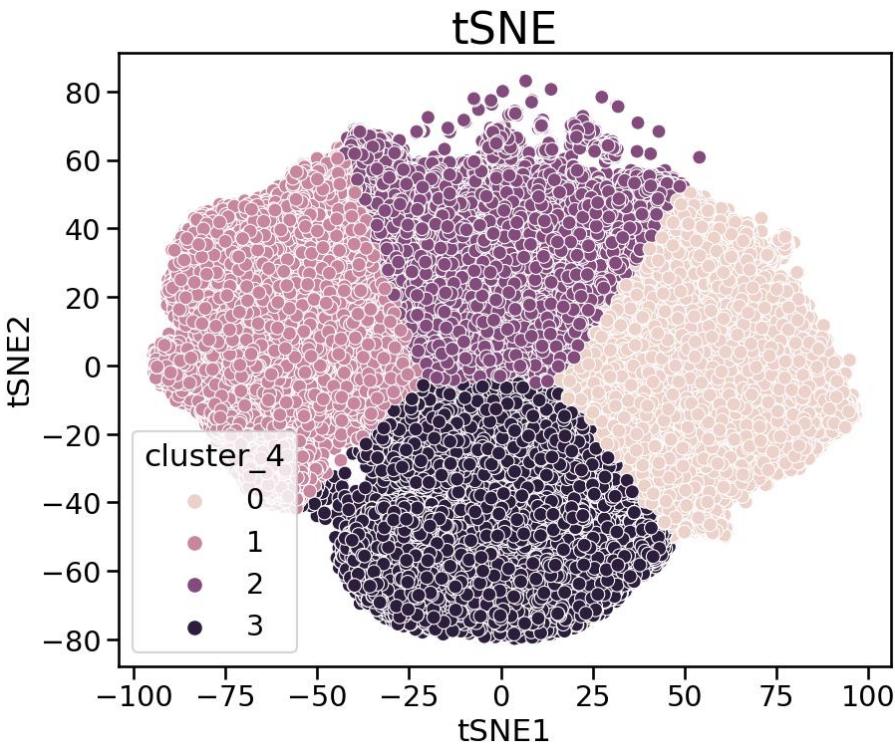


Figure 5.31: scatterplot K-means (4 clusters) con dati ridotti t-SNE

Anche in questo caso, è possibile osservare che la divisione effettuata dal clustering porta alla regione molto fitta di osservazioni divisa in quattro parti abbastanza bilanciate, assegnando i pochissimi valori lontani dalla “nuvola” principale al cluster più vicino.

	Silhouette coefficient [-1,1]	Calinski-Harabasz index [0, +∞]	Davies-Bouldin index [0, +∞]	Dunn index [0, +∞]	Entropy [0, log2(K)]
K-means(4 clusters)	0.404	285871.884	0.806	0.247	12.162

Table 5.30: indici K-means (4 clusters) con dati ridotti t-SNE

Per quanto riguarda le metriche considerate, il valore dell’indice di Silhouette è piuttosto basso (vicino allo 0.5), sinonimo di clusters abbastanza ben definiti e abbastanza separati gli uni dagli altri; l’indice di Calinski-Harabasz, invece, assume valore molto elevato (anche più del K-means con due clusters), sinonimo di clusters ben separati ed internamente molto densi; il valore dell’indice di Davies-Bouldin invece è basso, ad indicare un’ottima separazione dei clusters; il valore dell’indice di Dunn è tendenzialmente basso, come in tutti i casi precedenti; infine, l’entropia assume valore pari a 12.162. Anche in questo caso i valori sono tendenzialmente in linea con quanto visto nel caso dell’UMAP e della PCA come metodi di riduzione della dimensionalità.

Infine, nel quarto ed ultimo caso, il cluster 0 ha numerosità 54394, il cluster 1 è composto da 74976 osservazioni, il cluster 2 è composto da 43652 osservazioni e il cluster 3 è composto da 61429 osservazioni, e il cluster 4 è composto da 51263 osservazioni.

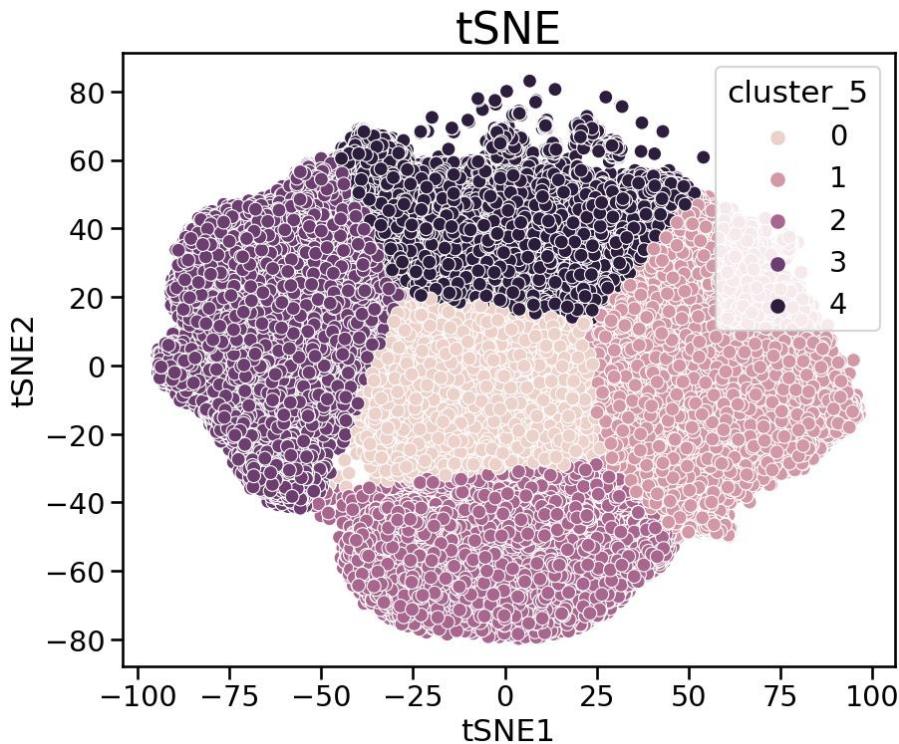


Figure 5.32: scatterplot K-means (5 clusters) con dati ridotti t-SNE

Con l'ultima suddivisione creata dal clustering, la regione molto fitta di osservazioni viene divisa in cinque parti (una regione al centro e quattro zone laterali), assegnando i pochissimi valori lontani dalla “nuvola” principale al cluster più vicino.

	Silhouette coefficient [-1,1]	Calinski-Harabasz index [0, +∞]	Davies-Bouldin index [0, +∞]	Dunn index [0, +∞]	Entropy [0, log2(K)]
K-means(5 clusters)	0.404	266677.214	0.843	0.173	12.221

Table 5.31: indici K-means (5 clusters) con dati ridotti t-SNE

Per quanto riguarda le metriche considerate, il valore dell’indice di Silhouette è piuttosto basso (vicino allo 0.5), sinonimo di clusters abbastanza ben definiti e abbastanza separati gli uni dagli altri; l’indice di Calinski-Harabasz, invece, assume valore molto elevato, sinonimo di clusters ben separati ed internamente molto densi; il valore dell’indice di Davies-Bouldin invece è basso, ad indicare un’ottima separazione dei clusters; anche in quest’ultimo caso, il valore dell’indice di Dunn è tendenzialmente basso; infine, l’entropia assume valore pari a 12.221. Anche in questo caso i valori sono tendenzialmente in linea con quanto visto nel caso dell’UMAP e della PCA come metodi di riduzione della dimensionalità.

5.2 Analisi miglior risultato di cluster analysis ottenuto

A fronte di un confronto dei risultati con gli esperti di dominio, e col lavoro svolto all'interno del gruppo di lavoro del MUDI Lab, è stato scelto come miglior risultato del clustering quello ottenuto col metodo di clustering *K-means* per due cluster, in seguito a riduzione di dimensionalità col metodo *UMAP(n_neighbors = 15, min_dist = 0.1)*, primo risultato del paragrafo [5.1.1].

Infatti, è risultato essere tra i metodi con valori migliori del coefficiente di Silhouette, considerato il più importante tra gli indici calcolati, oltre a produrre clusters di dimensioni significative (il cluster 0 ha numerosità 213931, il cluster 1 è composto da 58183 osservazioni). Inoltre, essendo due soli clusters, sarebbe possibile confrontarlo successivamente con una serie di aspetti fondamentali sia già presenti all'interno del dataset (condizioni del parto, come *Twins*, *AntibioticsMother*, *CortisoneBaby* o *Premature*, o aspetti demografici, come *Sex*, fino alla provenienza o meno da determinati reparti per la variabile *Reparto* o il confronto con soglie significative di variabili quantitative), sia per eventuali dati futuri sullo sviluppo o meno di determinate malattie metaboliche.

5.2.1 Statistiche descrittive variabili quantitative stratificate per cluster

Osserviamo ora le statistiche descrittive delle variabili quantitative stratificate in base ai due clusters ottenuti col metodo di clustering *K-means* per due clusters, in seguito a riduzione di dimensionalità col metodo *UMAP(n_neighbors = 15, min_dist = 0.1)*. La seguente tabella riporta le statistiche relative alle osservazioni assegnate al cluster 0 (213931 osservazioni).

	min	2.5%	25%	50%	75%	97.5%	max	iqr	skew	kurt	mean	std	outlier%
Ala	0.0	142.63	211.17	258.3	318.87	487.95	998529.91	107.7	103.47	13996.84	335.95	4951.29	5%
Arg	0.0	1.53	5.14	8.92	14.7	35.84	5352.47	9.56	101.94	12649.73	11.72	32.57	5%
Cit	0.0	6.02	10.9	13.97	17.83	29.39	6032.4	6.93	107.96	14668.6	15.35	34.14	5%
Gly	0.0	172.16	307.84	390.52	485.86	760.48	926743.38	178.02	69.38	5277.49	545.24	8515.01	5%
Leu\Ile\Pro-OH	0.0	87.82	122.3	146.14	176.62	263.05	317359.92	54.32	77.63	6385.02	191.85	2756.28	5%
Orn	0.0	50.18	79.91	102.18	132.5	235.97	287352.74	52.59	126.67	18429.07	127.6	1563.69	5%
MET	0.0	10.24	15.88	19.59	23.91	36.23	9288.54	8.03	97.24	11628.63	21.21	52.2	5%
PHE	0.12	33.53	45.75	53.22	61.98	86.26	132111.06	16.23	222.3	51455.6	58.68	538.28	5%
TYR	0.0	44.77	70.54	88.97	113.22	192.03	331329.89	42.68	149.07	26390.52	109.69	1470.05	5%
Pro	0.0	108.77	150.33	176.86	209.13	299.68	400411.1	58.8	69.95	5410.75	237.22	3386.14	5%
Val	0.0	74.14	108.95	131.64	159.12	232.81	760742.12	50.17	140.55	26238.05	172.73	3173.48	5%
C0	0.0	8.16	13.87	18.5	24.44	42.0	5245.8	10.57	68.8	5794.26	20.76	38.81	5%
C3	0.0	0.44	1.15	1.71	2.36	4.29	77.52	1.21	9.72	480.02	1.86	1.13	5%
C4OH\C3DC	0.0	0.05	0.1	0.16	0.23	0.39	3.63	0.13	1.45	14.19	0.17	0.09	5%
C4	0.0	0.09	0.16	0.21	0.28	0.56	11.98	0.12	8.0	471.99	0.23	0.13	5%
C5OH\C4DC	0.0	0.09	0.15	0.19	0.24	0.36	26.76	0.09	99.9	23254.89	0.2	0.1	4%
C5	0.0	0.05	0.09	0.11	0.15	0.29	10.92	0.06	19.74	2175.07	0.13	0.07	3%
C5:1	0.0	0.0	0.01	0.01	0.01	0.02	0.4	0.0	5.75	160.01	0.01	0.01	1%
C5DC\ C6OH	0.0	0.05	0.09	0.11	0.15	0.23	30.29	0.06	212.52	68695.36	0.12	0.09	5%
C6	0.0	0.02	0.03	0.04	0.05	0.09	3.28	0.02	19.8	2463.84	0.05	0.02	3%
C6DC	0.0	0.04	0.08	0.11	0.15	0.24	15.53	0.07	73.89	18264.93	0.12	0.06	4%
C8	0.0	0.02	0.04	0.05	0.07	0.13	33.89	0.03	296.43	111730.41	0.06	0.09	3%
C8:1	0.0	0.01	0.02	0.03	0.05	0.17	2.37	0.03	4.35	70.07	0.04	0.04	3%
C10	0.0	0.03	0.06	0.08	0.11	0.21	2.29	0.05	3.64	56.5	0.09	0.05	3%
C10:1	0.0	0.02	0.04	0.05	0.06	0.09	1.12	0.02	4.09	90.28	0.05	0.02	3%
C10:2	0.0	0.0	0.0	0.01	0.01	7.66	0.01	387.32	168408.29	0.0	0.02	2%	
C12	0.0	0.03	0.06	0.09	0.14	0.28	11.18	0.08	18.89	2574.61	0.11	0.07	4%
C12:1	0.0	0.02	0.04	0.06	0.1	0.23	1.5	0.06	2.29	13.43	0.08	0.06	4%
C14	0.0	0.06	0.15	0.2	0.27	0.43	7.24	0.12	4.18	192.92	0.21	0.1	4%
C14:1	0.0	0.02	0.06	0.1	0.14	0.29	11.42	0.08	17.97	2124.41	0.11	0.08	4%
C14:2	0.0	0.01	0.01	0.02	0.02	0.04	0.71	0.01	6.73	310.62	0.02	0.01	3%
C14-OH	0.0	0.0	0.01	0.01	0.02	0.04	0.32	0.01	1.99	20.05	0.02	0.01	1%
C16	0.0	0.51	2.11	3.15	4.1	6.32	75.53	1.99	3.47	99.9	3.13	1.65	5%
C16:1	0.0	0.02	0.14	0.22	0.29	0.45	11.03	0.15	4.01	320.98	0.21	0.12	3%

C16-OH	0.0	0.01	0.01	0.02	0.03	0.05	2.34	0.02	59.38	9495.5	0.02	0.01	3%
C16:1-OH	0.0	0.01	0.03	0.04	0.05	0.08	0.61	0.02	1.31	20.51	0.04	0.02	1%
C18	0.0	0.27	0.68	0.94	1.21	1.89	32.85	0.53	8.53	382.76	0.97	0.46	5%
C18:1	0.0	0.56	1.16	1.52	1.93	2.91	43.34	0.77	8.93	367.67	1.58	0.68	5%
C18:2	0.0	0.06	0.12	0.16	0.23	0.48	11.14	0.12	9.88	628.39	0.19	0.12	5%
C18-OH	0.0	0.0	0.01	0.01	0.02	0.03	3.38	0.01	137.56	39912.85	0.01	0.01	1%
C18:1-OH	0.0	0.01	0.02	0.02	0.03	0.05	1.59	0.01	26.1	2910.16	0.02	0.01	3%
SA	0.0	0.22	0.57	0.78	1.02	1.51	21.48	0.45	7.65	271.0	0.8	0.39	5%
GestationalAge	23.0	33.0	38.0	39.0	40.0	41.0	43.0	2.0	-2.16	7.72	38.54	2.16	2%
Weight	350.0	1722.25	2880.0	3220.0	3535.0	4130.0	5000.0	655.0	-0.9	1.92	3162.03	582.01	5%

Table 5.32: tabella informazioni e statistiche descrittive variabili quantitative cluster 0

La seguente tabella riporta invece le statistiche relative alle osservazioni assegnate al cluster 1 (58183 osservazioni).

	min	2.5%	25%	50%	75%	97.5%	max	iqr	skew	kurt	mean	std	outlier%
Ala	0.0	142.47	210.73	258.6	318.85	487.58	475231.04	108.12	63.15	4230.59	365.09	5407.44	5%
Arg	0.0	1.53	5.08	8.85	14.61	35.68	4024.69	9.53	86.79	9195.52	11.66	30.35	5%
Cit	0.0	6.04	10.89	13.99	17.82	29.29	5411.04	6.93	124.54	19258.87	15.25	30.47	5%
Gly	0.0	173.19	308.72	390.79	484.87	749.75	656552.86	176.16	89.01	8405.38	473.98	5488.52	5%
Leu\Ile\Pro-OH	0.0	87.94	122.11	145.85	176.32	261.37	348341.42	54.21	71.61	5727.59	204.07	3193.4	5%
Orn	0.0	50.31	79.66	101.76	132.27	233.85	273155.95	52.61	103.08	11585.56	132.97	1908.4	5%
MET	0.0	10.32	15.85	19.53	23.87	36.23	2823.1	8.02	79.74	7087.34	20.81	27.39	5%
PHE	12.02	33.6	45.76	53.26	61.83	85.93	140135.98	16.07	231.4	54950.47	58.88	589.1	5%
TYR	0.08	44.5	70.47	88.75	112.77	190.92	352349.87	42.3	159.78	26587.97	110.42	1942.69	5%
Pro	0.0	109.16	149.96	176.2	208.04	298.24	527548.27	58.08	107.03	13917.26	222.22	3267.79	5%
Val	0.0	74.41	108.95	131.49	158.67	232.31	300268.94	49.72	64.47	4516.06	193.12	3255.19	5%
C0	0.0	8.13	13.83	18.44	24.46	41.97	7978.04	10.63	107.62	15312.92	20.73	47.28	5%
C3	0.0	0.44	1.15	1.71	2.36	4.3	55.16	1.21	5.5	164.01	1.86	1.09	5%
C4OHIC3DC	0.0	0.05	0.1	0.16	0.23	0.39	1.41	0.13	1.11	2.61	0.17	0.09	5%
C4	0.0	0.08	0.16	0.2	0.28	0.56	10.87	0.12	16.0	1048.0	0.23	0.14	4%
C5OHIC4DC	0.0	0.09	0.15	0.19	0.24	0.36	3.37	0.09	3.04	74.95	0.2	0.07	4%
C5	0.0	0.05	0.08	0.11	0.14	0.29	10.86	0.06	42.39	5273.92	0.12	0.08	3%
C5:1	0.0	0.0	0.01	0.01	0.01	0.02	0.23	0.0	5.41	102.68	0.01	0.01	1%
C5DC\C6OH	0.0	0.05	0.08	0.11	0.15	0.23	7.62	0.06	48.34	5320.19	0.12	0.07	5%
C6	0.0	0.02	0.03	0.04	0.05	0.09	1.65	0.02	9.48	613.39	0.05	0.02	3%
C6DC	0.0	0.04	0.08	0.11	0.15	0.24	1.47	0.07	1.44	15.21	0.12	0.05	4%
C8	0.0	0.02	0.04	0.05	0.07	0.13	4.62	0.03	32.86	3447.67	0.06	0.04	3%
C8:1	0.0	0.01	0.02	0.03	0.05	0.17	0.99	0.03	3.58	24.25	0.05	0.04	3%
C10	0.0	0.03	0.06	0.08	0.11	0.21	1.38	0.05	2.9	25.6	0.09	0.05	4%
C10:1	0.0	0.02	0.04	0.05	0.06	0.09	0.79	0.02	4.02	74.83	0.05	0.02	3%
C10:2	0.0	0.0	0.0	0.01	0.02	0.35	0.01	8.05	335.25	0.0	0.01	2%	
C12	0.0	0.03	0.06	0.09	0.14	0.28	1.03	0.08	1.86	6.48	0.11	0.07	4%
C12:1	0.0	0.02	0.04	0.06	0.1	0.23	0.96	0.06	2.1	8.85	0.08	0.06	5%
C14	0.0	0.06	0.15	0.2	0.27	0.43	2.01	0.12	0.92	4.26	0.21	0.1	5%
C14:1	0.0	0.02	0.06	0.1	0.14	0.29	2.4	0.08	2.12	22.01	0.11	0.07	4%
C14:2	0.0	0.01	0.01	0.02	0.02	0.04	0.62	0.01	10.0	457.4	0.02	0.01	3%
C14-OH	0.0	0.0	0.01	0.01	0.02	0.04	0.45	0.01	3.1	78.16	0.02	0.01	1%
C16	0.0	0.51	2.11	3.15	4.1	6.34	65.46	1.99	3.66	104.74	3.13	1.66	5%
C16:1	0.0	0.02	0.14	0.22	0.29	0.45	1.97	0.15	0.35	0.94	0.21	0.12	3%
C16-OH	0.0	0.01	0.01	0.02	0.03	0.05	1.92	0.02	50.32	5808.96	0.02	0.01	4%
C16:1-OH	0.0	0.01	0.03	0.04	0.05	0.08	0.86	0.02	2.83	101.94	0.04	0.02	1%
C18	0.0	0.27	0.69	0.94	1.21	1.89	21.96	0.52	5.92	200.36	0.97	0.45	5%
C18:1	0.0	0.56	1.16	1.52	1.92	2.9	43.54	0.76	12.38	574.65	1.58	0.71	5%
C18:2	0.0	0.06	0.11	0.16	0.23	0.47	2.82	0.12	3.2	23.36	0.19	0.11	5%
C18-OH	0.0	0.0	0.01	0.01	0.02	0.03	0.71	0.01	17.81	1356.53	0.01	0.01	1%
C18:1-OH	0.0	0.01	0.02	0.02	0.03	0.04	1.38	0.01	30.21	2872.76	0.02	0.01	3%
SA	0.0	0.22	0.57	0.78	1.02	1.51	21.61	0.45	9.48	355.66	0.8	0.4	5%
GestationalAge	23.0	33.0	38.0	39.0	40.0	41.0	43.0	2.0	-2.18	7.74	38.55	2.17	2%
Weight	433.0	1720.0	2880.0	3220.0	3540.0	4140.0	5000.0	660.0	-0.9	1.91	3163.03	583.36	5%

Table 5.33: tabella informazioni e statistiche descrittive variabili quantitative cluster 1

Com'era lecito aspettarsi, osserviamo che generalmente i valori non variano sensibilmente al cambiare il cluster di appartenenza: le distribuzioni delle variabili quantitative sembrano mantenersi abbastanza costanti al variare il cluster di appartenenza, con alcune variazioni più o meno significative (osservando la media e la varianza, le features che variano maggiormente sono Ala e Gly, con variazioni rispettivamente di 30 e 70 rispetto alle medie; inoltre molte osservazioni hanno poca differenza in termini di medie per l'unità di misura e la loro scala, ma sono effettivamente differenti nei due clusters, come C18:1, C18, C16:1, C16, C14:1, C14, C6DC, C5, C3 e PHE, tutti tendenzialmente con distribuzioni con valori minori per il cluster 1). È possibile confermare le informazioni trovate anche con i grafici ottenuti, presenti nell'Appendice 3 [[Figure analisi esplorative stratificate per cluster](#)].

Anche dai risultati ottenuti con la riduzione di dimensionalità non erano stati trovati clusters ben divisi, separati e densi internamente, bensì nuvole di dati fortemente compatte con osservazioni lontane dal resto dei dati.

5.2.2 Variabili qualitative

Per quanto riguarda le variabili qualitative, sono state create le seguenti cross tabulazioni per verificare le distribuzioni delle variabili in base al cluster assegnato (per ogni variabile qualitativa, riportata la variabile *cluster* sulle colonne e le categorie della variabile qualitativa di interesse sulle righe; tra parentesi vengono indicate le frequenze relative):

		Grouped by cluster			
		Missing	Overall	0	1
AntibioticsBaby, n (%)	0.0		253444 (93.1)	54238 (93.2)	199206 (93.1)
	1.0		18670 (6.9)	3945 (6.8)	14725 (6.9)
ARTFeed, n (%)	0		246641 (90.6)	52753 (90.7)	193888 (90.6)
	1		25473 (9.4)	5430 (9.3)	20043 (9.4)
HUFee, n (%)	0		92220 (33.9)	19717 (33.9)	72503 (33.9)
	1		179894 (66.1)	38466 (66.1)	141428 (66.1)
MIXFeed, n (%)	0		214641 (78.9)	45817 (78.7)	168824 (78.9)
	1		57473 (21.1)	12366 (21.3)	45107 (21.1)
TooYoung, n (%)	0.0		269152 (98.9)	57530 (98.9)	211622 (98.9)
	1.0		2962 (1.1)	653 (1.1)	2309 (1.1)
TyrroidMother, n (%)	0.0		231863 (85.2)	49529 (85.1)	182334 (85.2)
	1.0		40251 (14.8)	8654 (14.9)	31597 (14.8)
Sex, n (%)	F		132257 (48.6)	28401 (48.8)	103856 (48.5)
	M		139857 (51.4)	29782 (51.2)	110075 (51.5)
TPNCARNFeed, n (%)	0		271794 (99.9)	58121 (99.9)	213673 (99.9)

	1		320 (0.1)	62 (0.1)	258 (0.1)
Meconium, n (%)	0.0	0	271712 (99.9)	58103 (99.9)	213609 (99.8)
	1.0		402 (0.1)	80 (0.1)	322 (0.2)
TPNFeed, n (%)	0	0	269219 (98.9)	57564 (98.9)	211655 (98.9)
	1		2895 (1.1)	619 (1.1)	2276 (1.1)
ENFeed, n (%)	0	0	271690 (99.8)	58089 (99.8)	213601 (99.8)
	1		424 (0.2)	94 (0.2)	330 (0.2)
AnswerIX, n (%)	1	0	268916 (98.8)	57516 (98.9)	211400 (98.8)
	2		3164 (1.2)	660 (1.1)	2504 (1.2)
	3		34 (0.0)	7 (0.0)	27 (0.0)
Reparto, n (%)	Generico	0	37373 (13.7)	7879 (13.5)	29494 (13.8)
	Neo-Pat		25726 (9.5)	5463 (9.4)	20263 (9.5)
	Nido		209015 (76.8)	44841 (77.1)	164174 (76.7)
CortisoneBaby, n (%)	0.0	0	260907 (95.9)	55741 (95.8)	205166 (95.9)
	1.0		11207 (4.1)	2442 (4.2)	8765 (4.1)
Etnia, n (%)	Arab	0	16182 (5.9)	3443 (5.9)	12739 (6.0)
	Asian		12016 (4.4)	2611 (4.5)	9405 (4.4)
	Black or African American		169 (0.1)	41 (0.1)	128 (0.1)
	Caucasian		225557 (82.9)	48152 (82.8)	177405 (82.9)
	Hispanic/Latino		6607 (2.4)	1427 (2.5)	5180 (2.4)
	Native Hawaiian or Other Pacific Islander		11574 (4.3)	2508 (4.3)	9066 (4.2)
	Other		9 (0.0)	1 (0.0)	8 (0.0)
TPNMCTFeed, n (%)	0	0	272087 (100.0)	58177 (100.0)	213910 (100.0)
	1		27 (0.0)	6 (0.0)	21 (0.0)

Table 5.34: distribuzione delle variabili qualitative stratificate per cluster

Osserviamo che generalmente, al variare del cluster di appartenenza, le proporzioni delle categorie rimangono costanti per ogni variabile qualitativa. Si osservano solo poche variazioni che riguardano piccole percentuali (scostamenti del 0.4% al massimo).

6. Conclusioni

Per quanto riguarda la riduzione di dimensionalità:

- Alcuni metodi sono computazionalmente pesanti (come kernel PCA, NMF, SVD...), dunque è possibile eseguire la riduzione di dimensionalità solo in alcune modalità, data la quantità di records contenuti nel dataset; sarebbe interessante testare metodi fuzzy, vista la natura dei dati (con in molti casi una nuvola di dati composta dalla grande maggioranza del dataset, e poche osservazioni sparse), tuttavia si tratta di metodi solitamente pesanti computazionalmente.
- Alcuni metodi, come l'UMAP e t-SNE, sono efficaci nel creare spazi di dimensioni inferiori significativi e con possibilità di personalizzazione dato dal tuning degli iperparametri; tuttavia, metodi come la t-SNE sono più leggeri e veloci nell'esecuzione dell'UMAP, e permettono di esplorare le possibilità con maggiore flessibilità.
- L'assenza di indici per valutare la bontà dei risultati ottenuti non permette di confrontare i risultati ottenuti con ogni risultato della riduzione di dimensionalità.

Per quanto riguarda la cluster analysis:

- Abbiamo osservato che alcuni metodi sono computazionalmente più pesanti (in particolare lo Spectral Clustering, ma anche il DBSCAN), mentre altri sono molto veloci e possono essere impiegati sull'intero dataset (come il metodo K-means, che impiega solamente pochi secondi sul campione da 50000 istanze).
- Alcuni metodi di cluster analysis fornivano come risultato dei clusters con dimensioni significative (come il clustering agglomerativo gerarchico col metodo di Ward, il BIRCH e il metodo delle k-medie), altri metodi (tutti quelli del clustering agglomerativo gerarchico ad eccezione del metodo di Ward, e il DBSCAN) invece tendevano a creare un grosso cluster che comprende la maggioranza delle osservazioni e tutti gli altri clusters formati da pochissime istanze.
- Dal punto di vista delle metriche, i valori migliori dell'indice di Silhouette erano assunti dal clustering gerarchico agglomerativo col metodo del legame medio, completo e singolo (in particolare per un numero basso di cluster ricercati, solitamente sotto i 10), il DBSCAN con valori intorno all'1; tutti gli altri metodi assumevano valori vicini allo 0, sinonimo di clusters non ben separati, densi internamente e definiti.
- Per quanto riguarda l'indice di Calinski-Harabasz, i valori migliori erano assunti dal metodo K-means, dal clustering gerarchico agglomerativo con metodo di Ward e dal BIRCH, mentre i valori più bassi erano assunti dal clustering gerarchico agglomerativo con cambio di metrica (quindi in caso di metrica l2, manhattan o coseno).
- I metodi migliori secondo l'indice di Davies-Bouldin erano nel clustering agglomerativo col metodo di Ward, nel BIRCH e nel K-means, mentre nel clustering gerarchico agglomerativo (ad eccezione di quello col metodo di Ward) era possibile vedere i valori più vicini allo 0.
- I valori dell'entropia generalmente crescevano all'aumentare del numero di cluster ricercato: in particolare, assumevano i valori (costantemente) più elevati per metodi quale il clustering gerarchico agglomerativo col metodo di Ward e col legame singolo e metrica diversa da quella euclidea, nel BIRCH e nelle k-medie, mentre assumevano valori più bassi (vicini allo 0) col DBSCAN; i valori dell'indice di Dunn, invece, si mantenevano generalmente bassi, intorno allo 0, lontano dalle condizioni ricercate.

Per quanto riguarda la cluster analysis applicata su dati a dimensionalità ridotta:

- Osserviamo che, utilizzando il metodo delle k-medie con 2 o 3 clusters (aspetto che si può analizzare ulteriormente, testando il comportamento con un numero più elevato di clusters ricercato), i gruppi di osservazioni hanno tutti dimensioni molto consistenti, formati da almeno 8000 istanze fino ad arrivare a 30000 circa.
- Tutte le metriche del clustering si sono comportate in maniera simile, a prescindere dal metodo di riduzione della dimensionalità utilizzato: l'indice di Silhouette ha assunto valori tendenzialmente bassi (ma più elevati delle k-medie senza riduzione di dimensionalità), l'indice di Calinski-Harabasz ha assunto valori molto elevati (sinonimo di clusters ben separati ed internamente molto densi), l'indice di Davies-Bouldin ha assunto valori vicini all'unità, mentre i valori dell'entropia si sono attestati intorno ai valori più elevati visti senza riduzione di dimensionalità.
- In generale, il metodo migliore sembra essere l'UMAP, che è il metodo in cui l'indice di Silhouette assume i valori più elevati. Tuttavia, è necessaria ulteriore osservazione dei risultati ottenuti, in quanto i clusters creati sembrano simili a livello di caratteristiche delle osservazioni ottenute: nell'analisi fatta sui clusters ottenuti con UMAP e K-means con 2 clusters, ci sono pochissime differenze come statistiche descrittive e composizioni dei clusters.

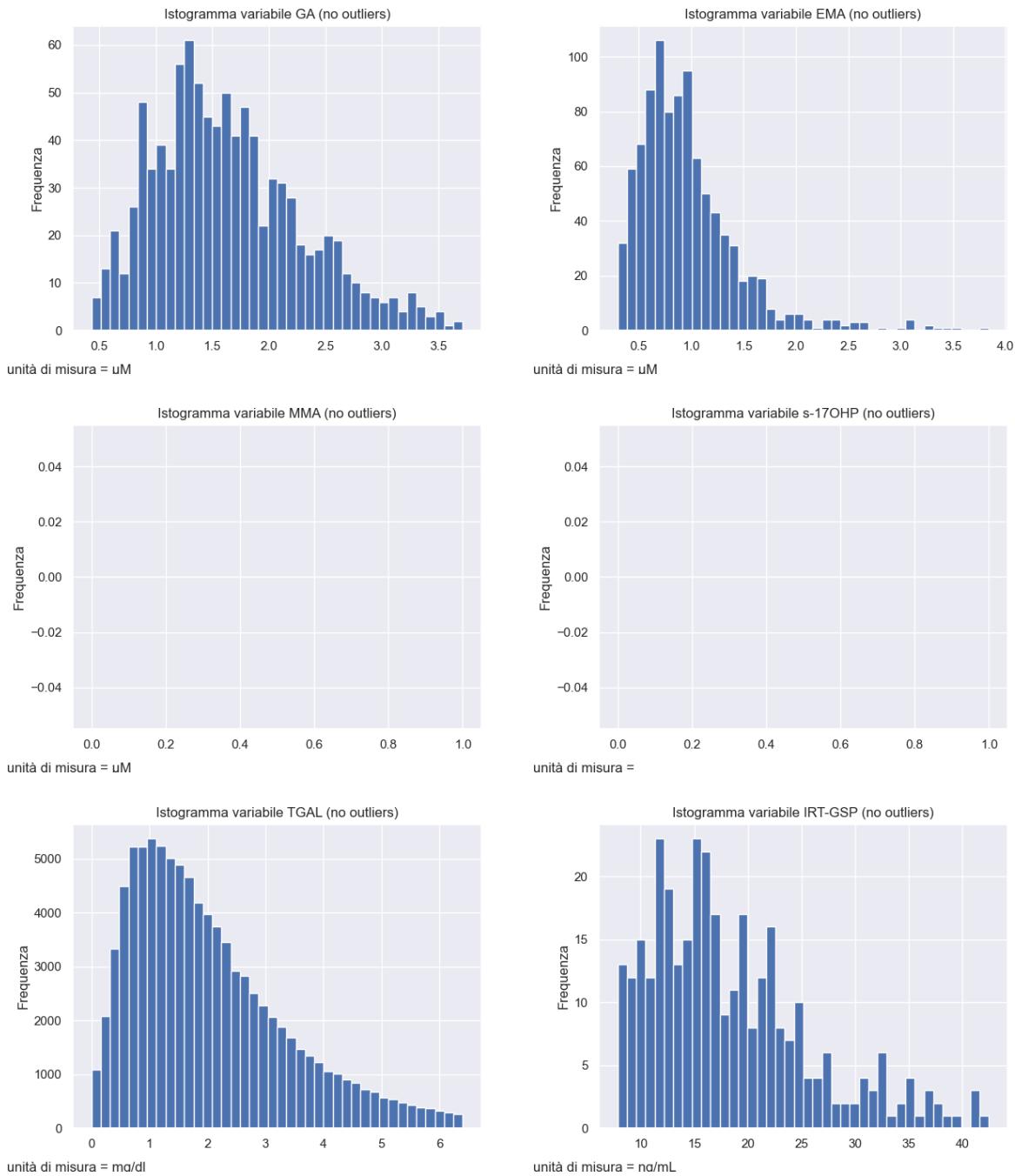
6.1.1 Sviluppi futuri

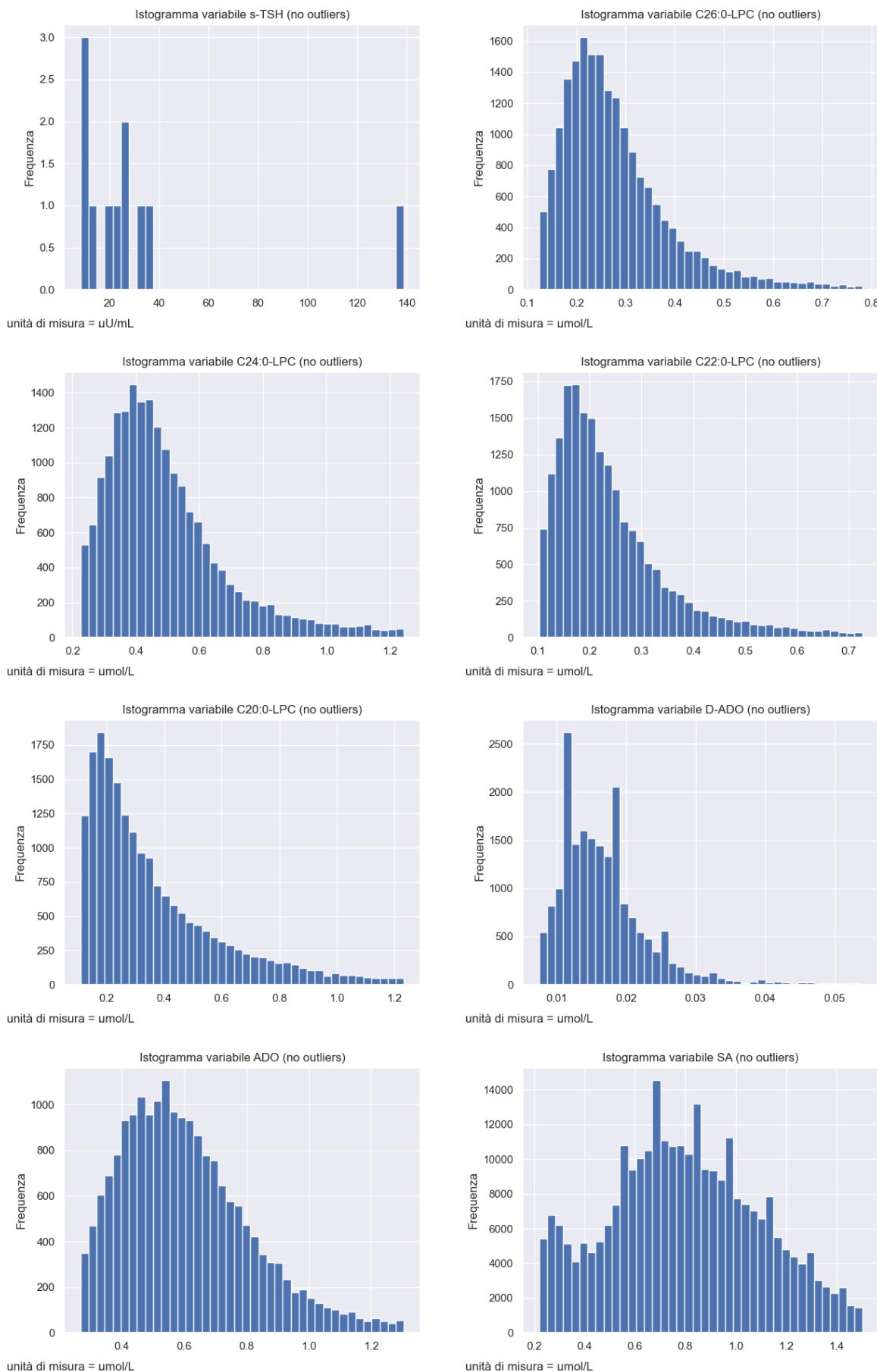
In questi mesi di lavoro è stato possibile rispondere a molte delle domande di ricerca emerse all'inizio del progetto e durante lo svolgimento stesso, tuttavia, ovviamente, sono sorte nuove ipotesi e questioni ancora da risolvere (a partire sia da suggerimenti degli esperti di dominio all'interno dell'ospedale Buzzi, che dalle informazioni ottenute grazie alle tecniche applicate e, non ultimo, dal confronto con la letteratura e gli studi pregressi in materia), che possono portare a nuovi sviluppi interessanti:

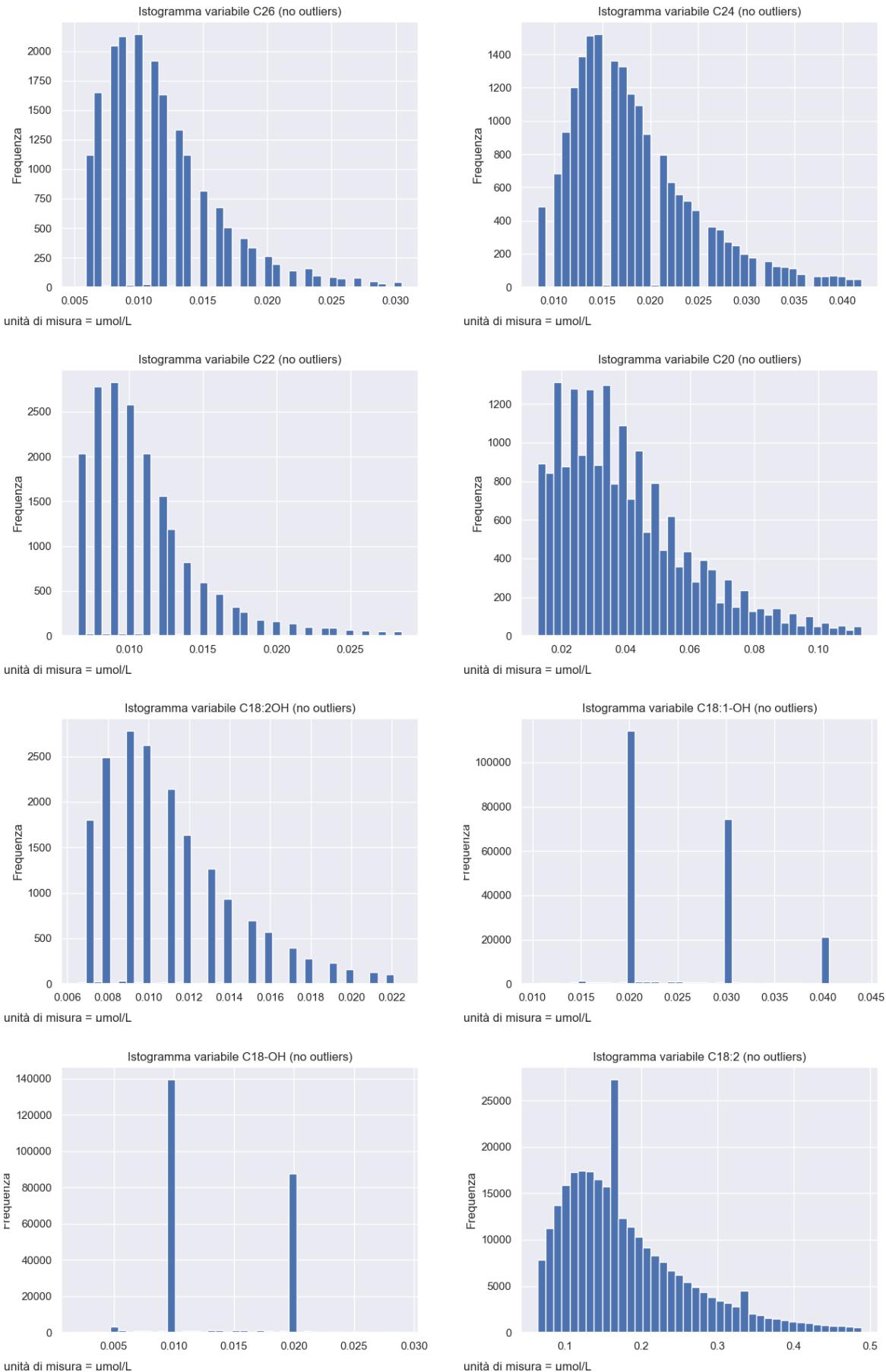
- Potrebbe essere interessante indagare ulteriormente i valori delle correlazioni tra alcune variabili qualitative particolarmente significative (ad esempio, analisi stratificate per *Etnia*) e variabili quantitative, e in generale valutare ulteriormente le analisi stratificate per la variabile *Reparto*.
- Potrebbe essere interessante studiare, come fatto con la variabile *Reparto*, le distribuzioni delle popolazioni rispetto ad altre variabili particolarmente significative, come *Etnia*, i tipi di alimentazione, le tipologie di parto (gemellare, prematuro...).
- Verificare se ci sono altri metodi utilizzabili per implementare lo Spectral Clustering, troppo dispendioso dal punto di vista computazionale per il caso di questo progetto.
- Verificare se ci sono metodi utilizzabili per implementare l'indice di Dunn in maniera da essere meno dispendioso dal punto di vista computazionale (in alcuni casi impossibile da calcolare per dati completi).
- Verificare altri metodi di cluster analysis efficaci per il caso in studio (ad esempio, metodi di clustering fuzzy)
- Verificare altre tecniche di riduzione della dimensionalità più efficaci.
- In caso sia possibile stabilire in maniera categorica la tecnica di cluster analysis più efficace, creare un'analisi, stratificata per clusters ottenuti, approfondita come le precedenti.
- In caso di disponibilità di dati relativi all'insorgenza o meno di malattie metaboliche per i neonati del dataset Buzzi, creare, implementare e testare modelli predittivi, e confrontare eventualmente i risultati anche con gli algoritmi di cluster analysis applicati, in modo da verificare se alcune tecniche sono particolarmente efficaci nel dividere gruppi di pazienti con particolari patologie metaboliche.

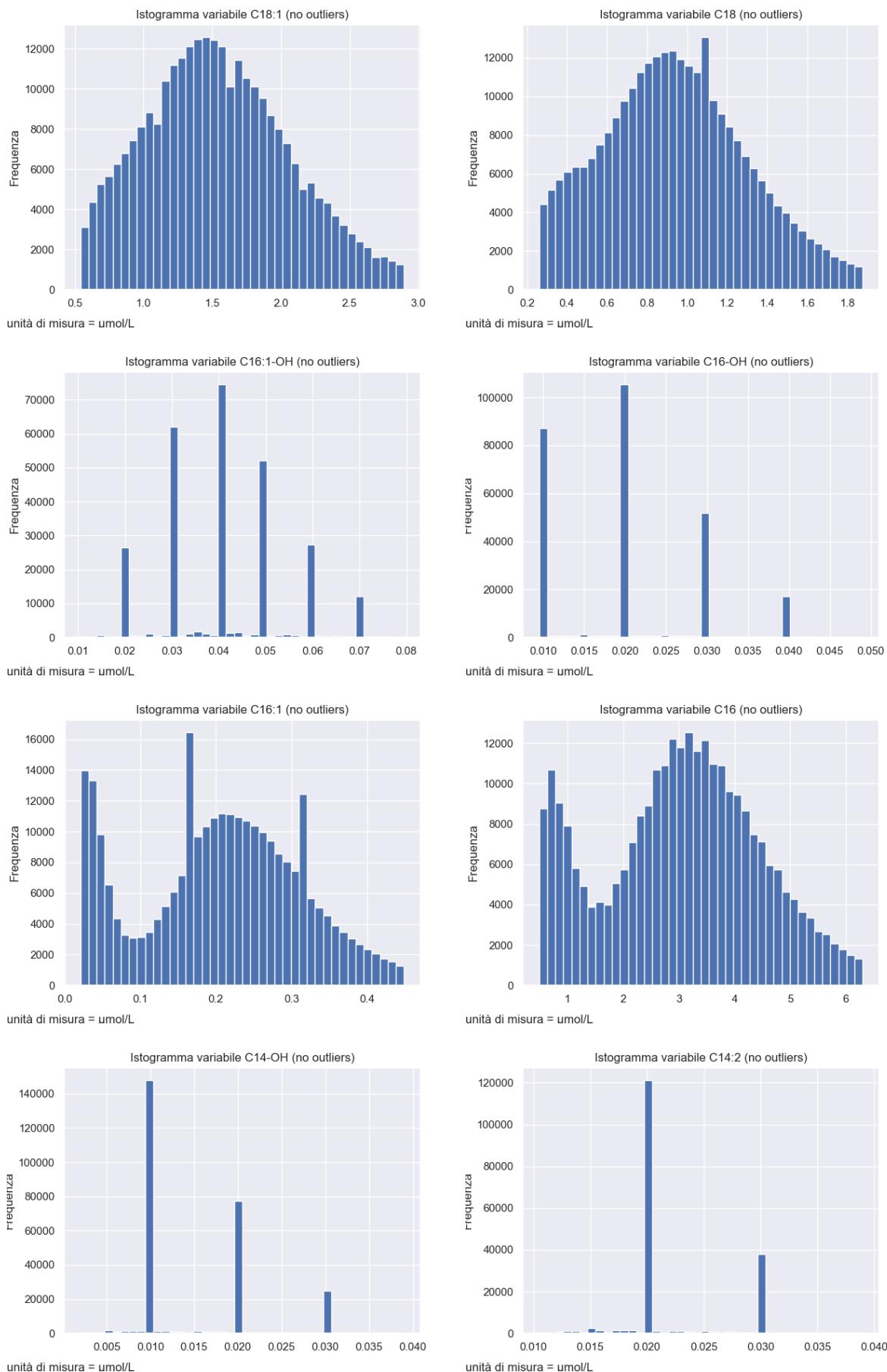
Appendice

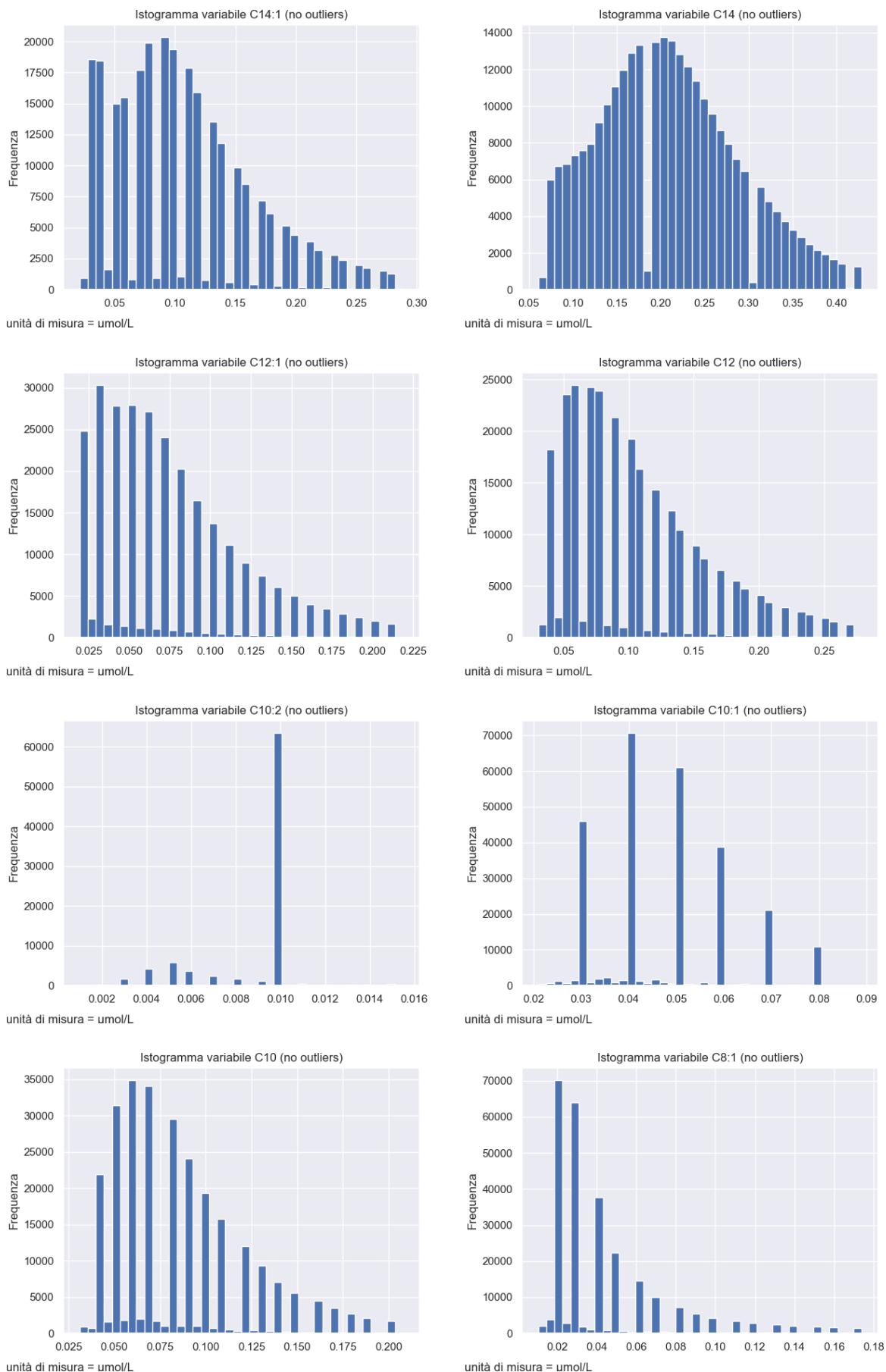
Figure analisi esplorative

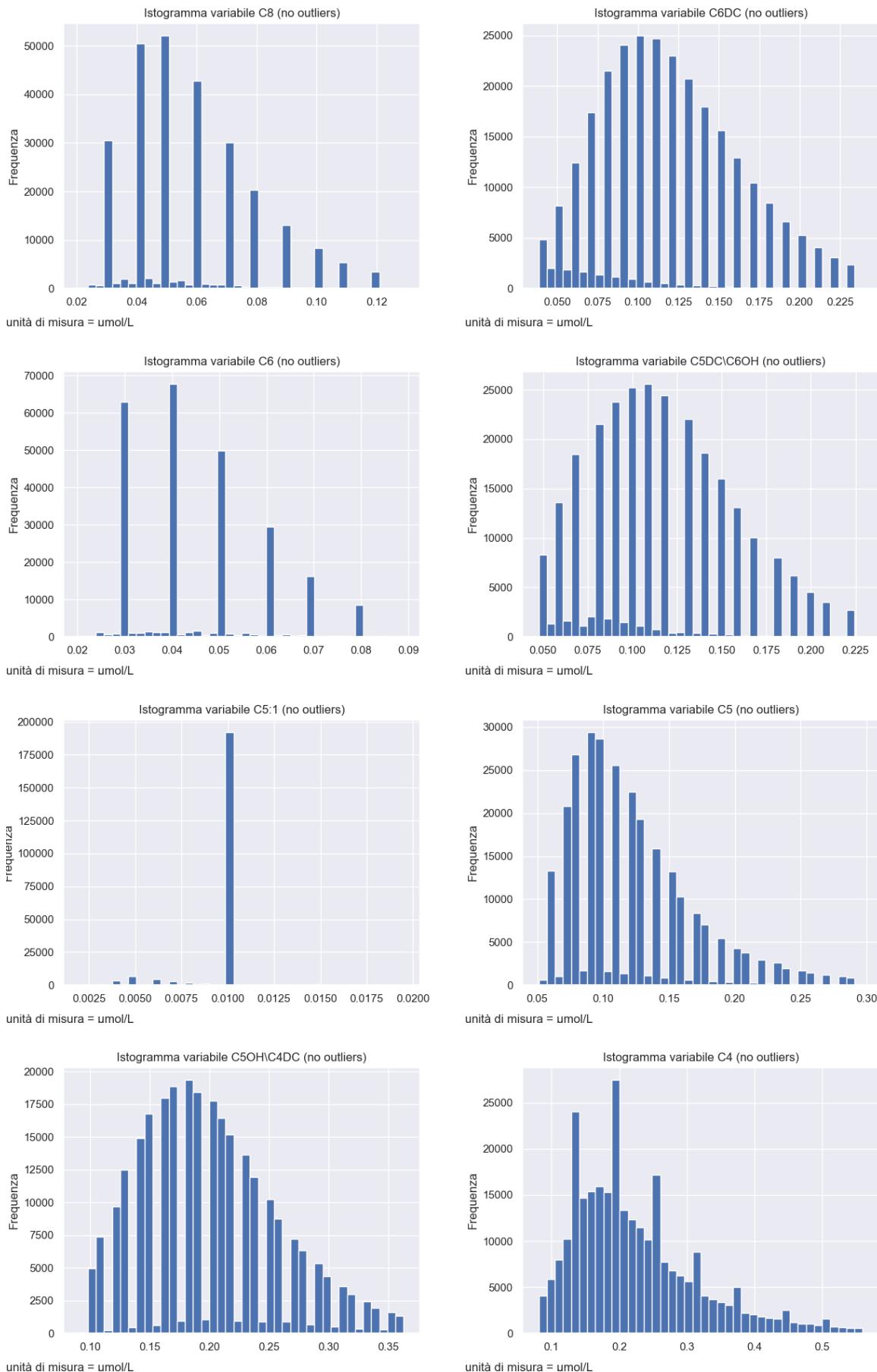


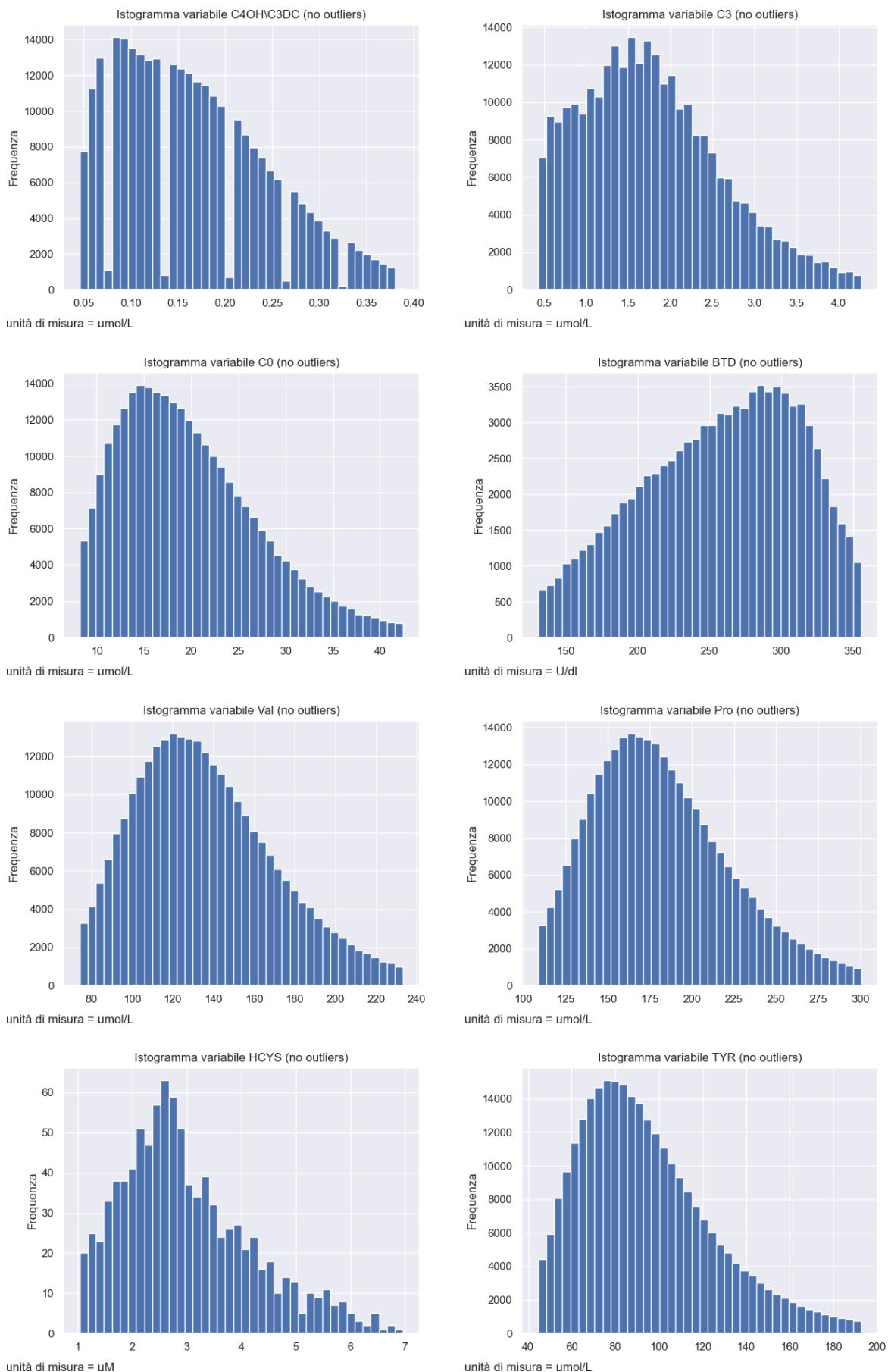


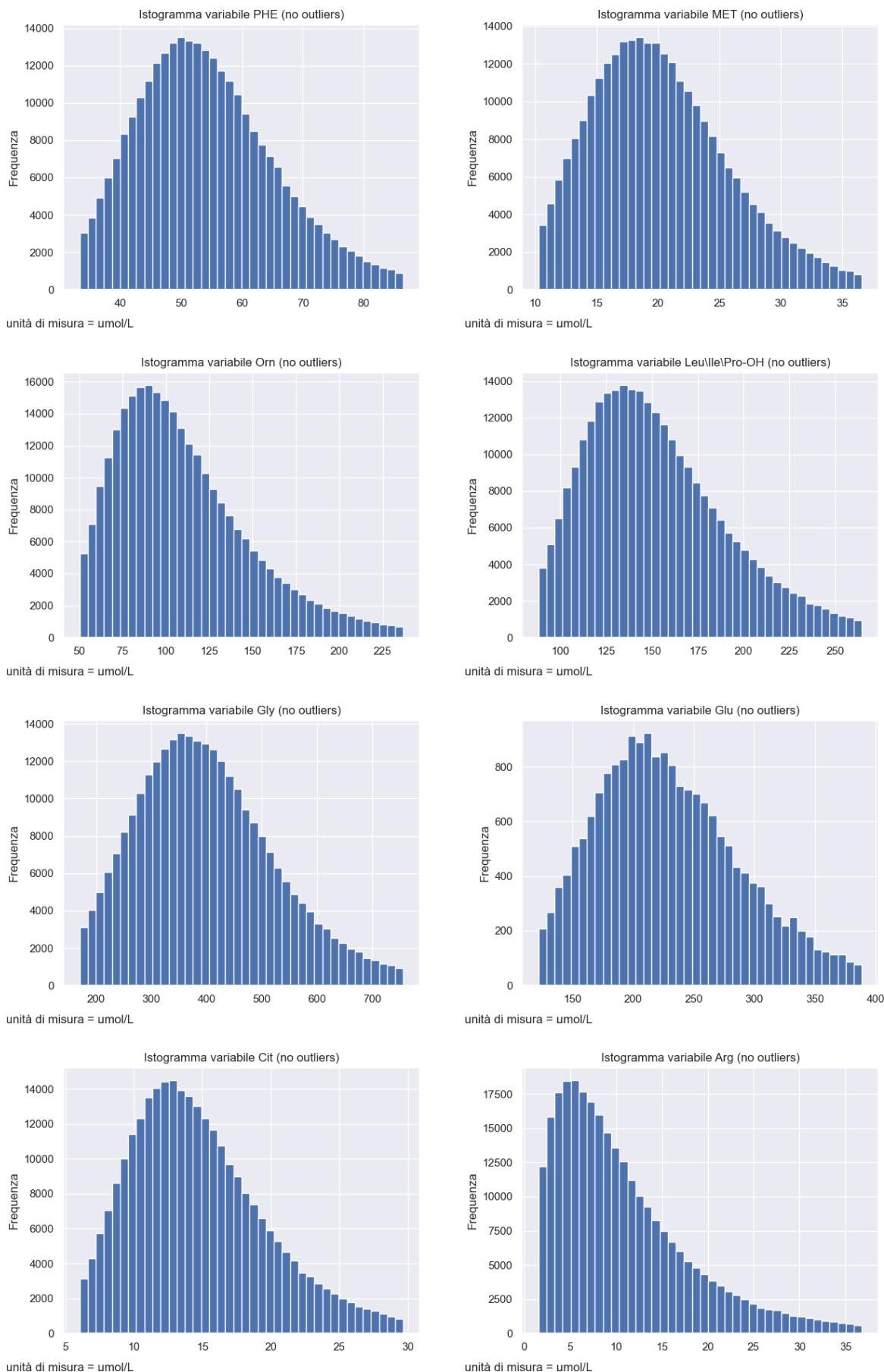


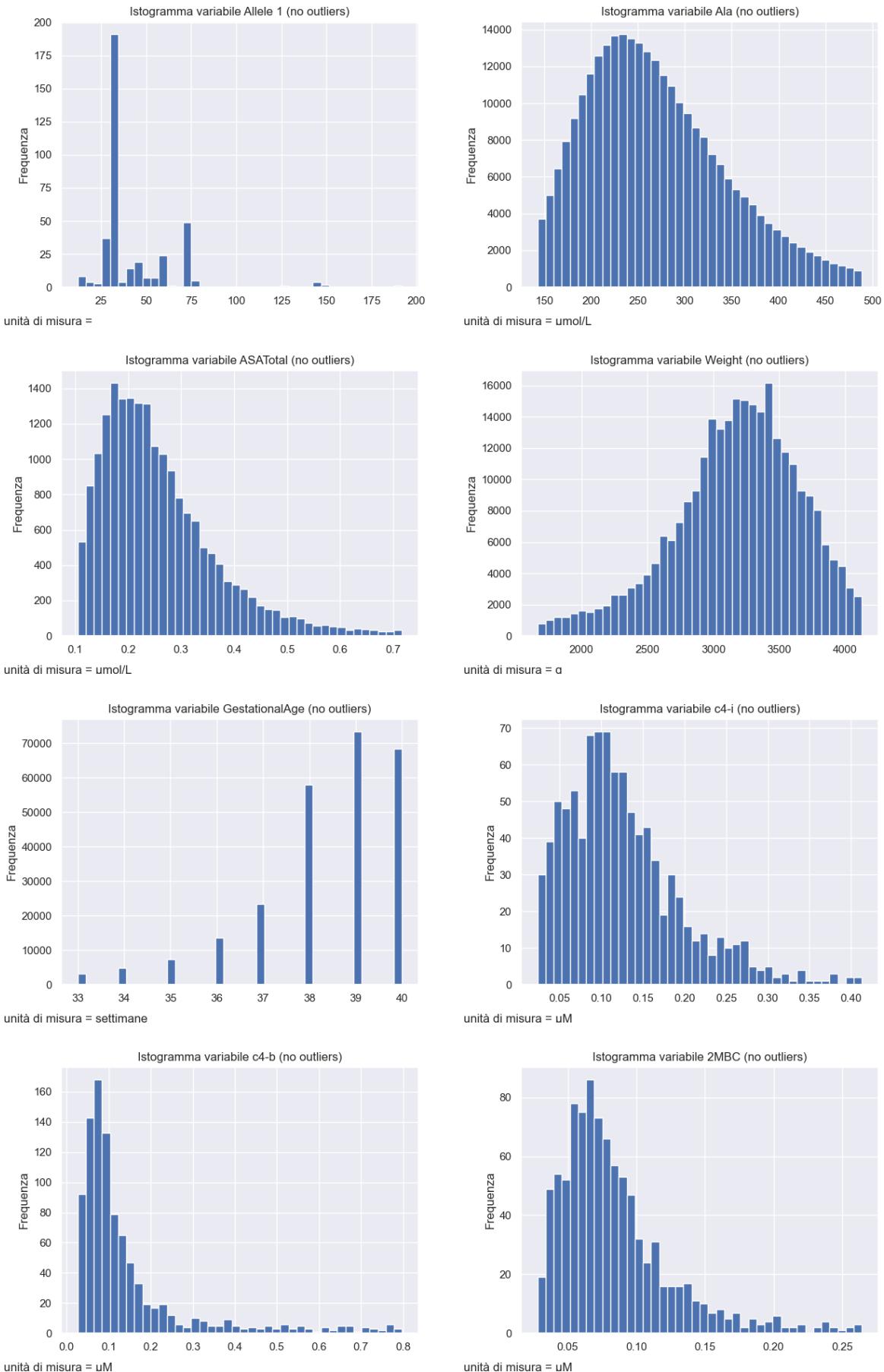












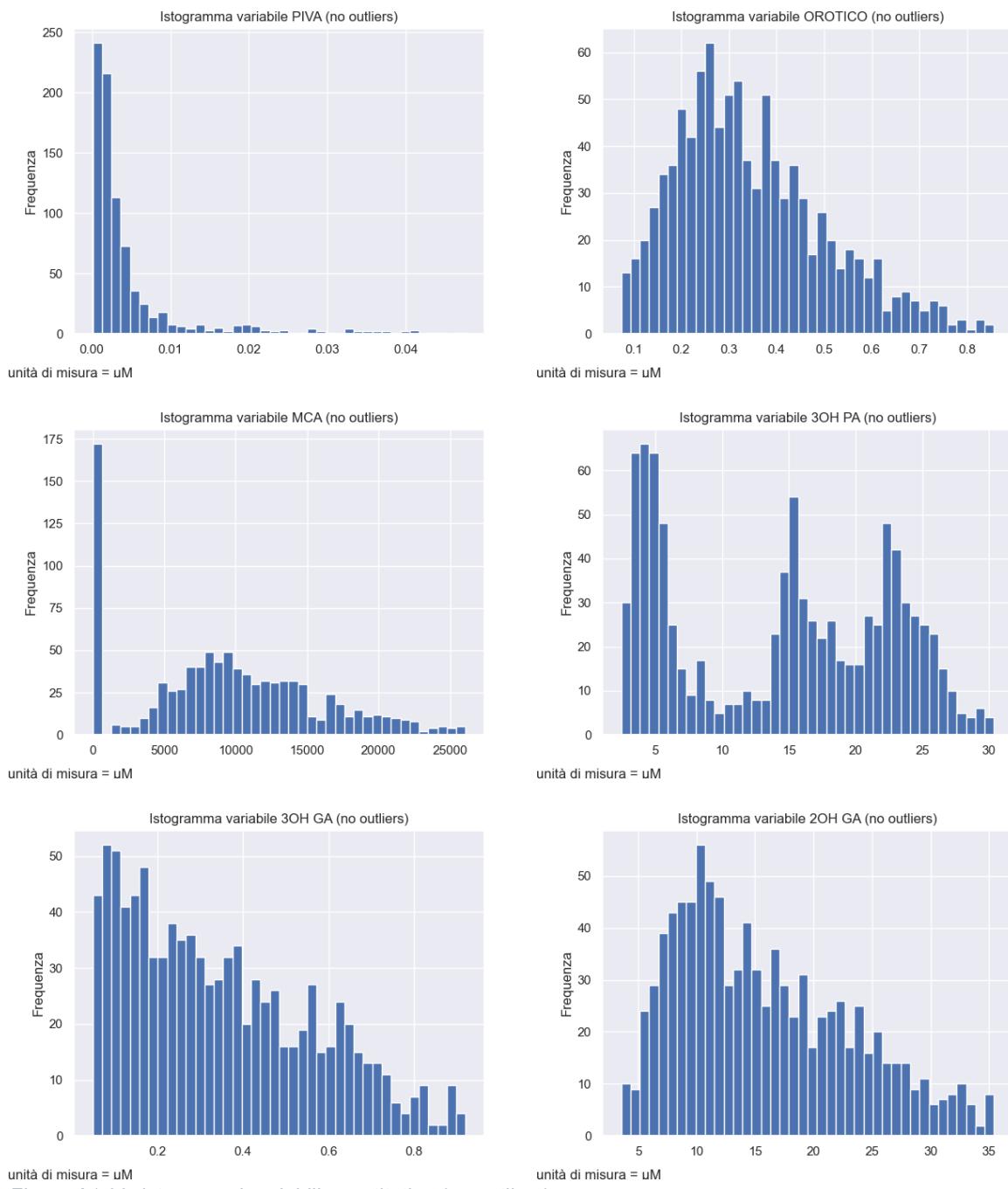
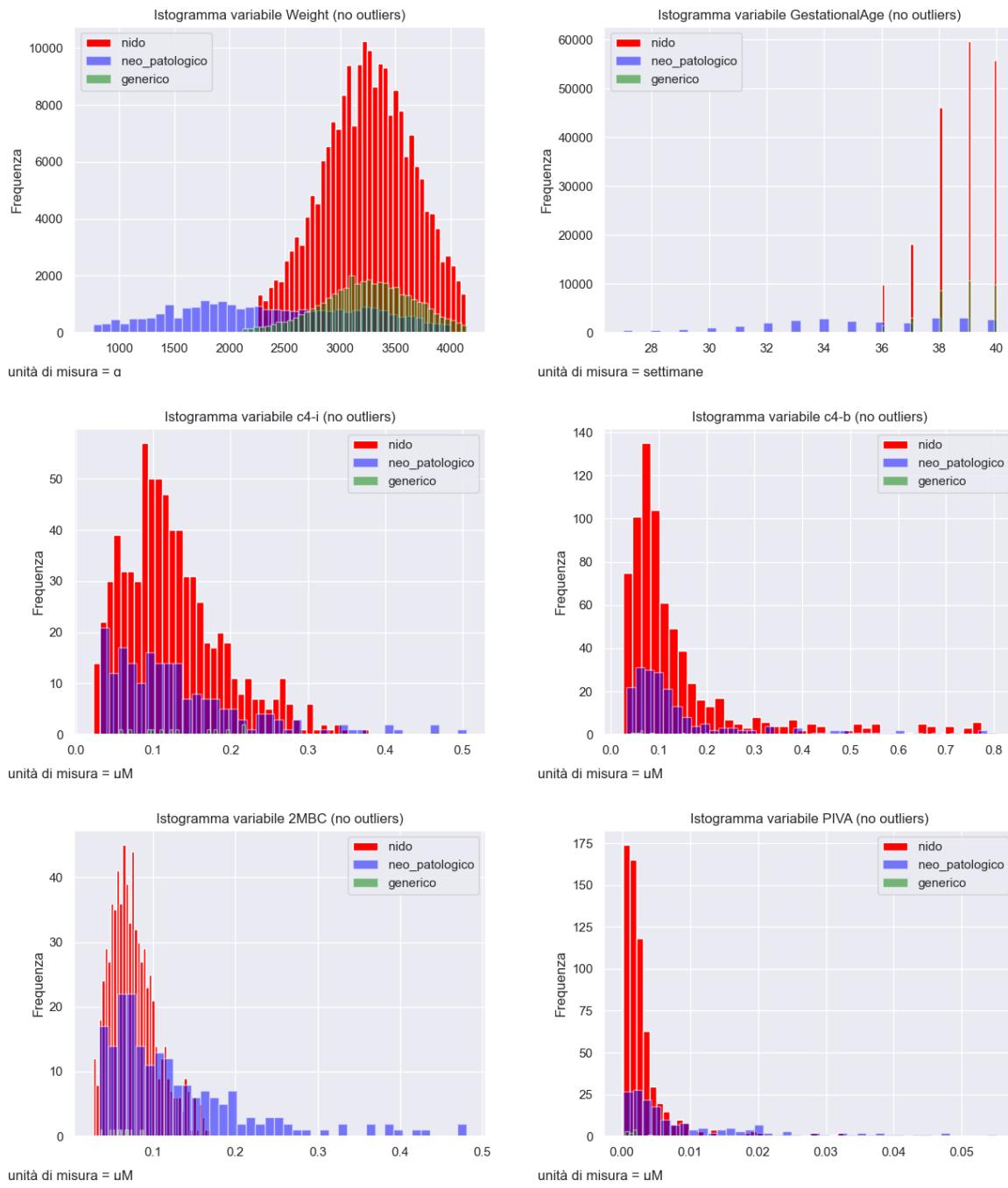
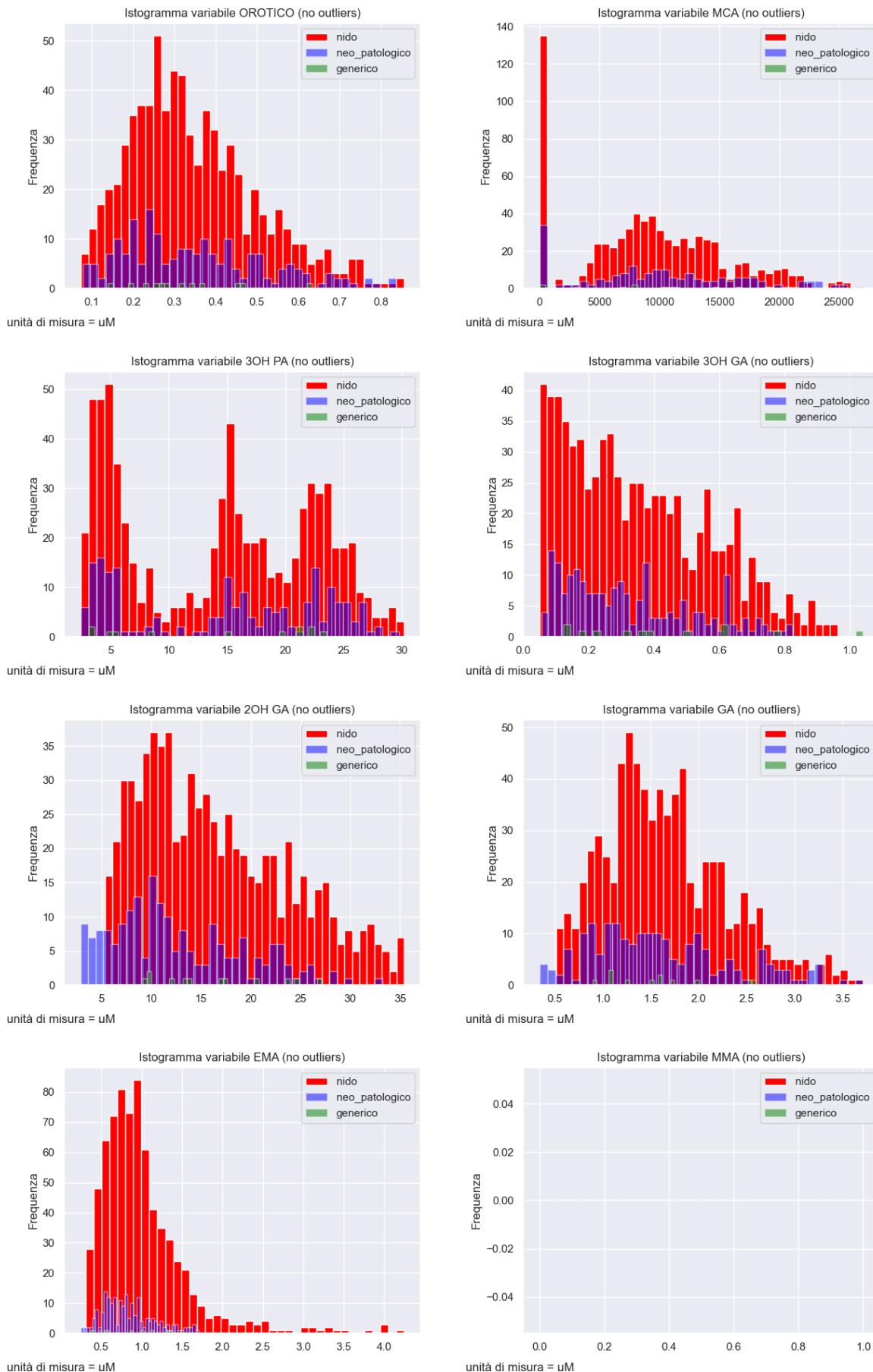
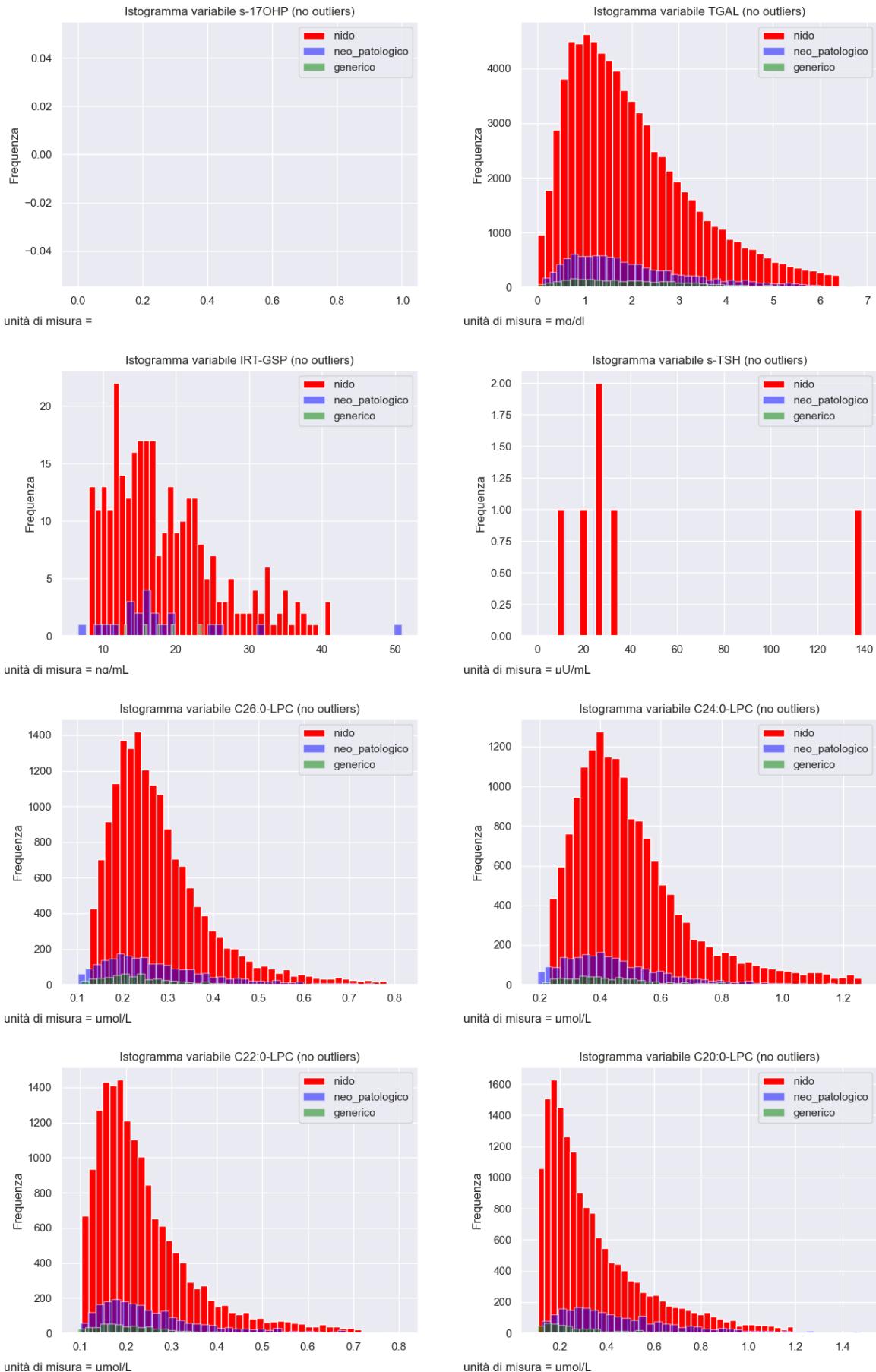


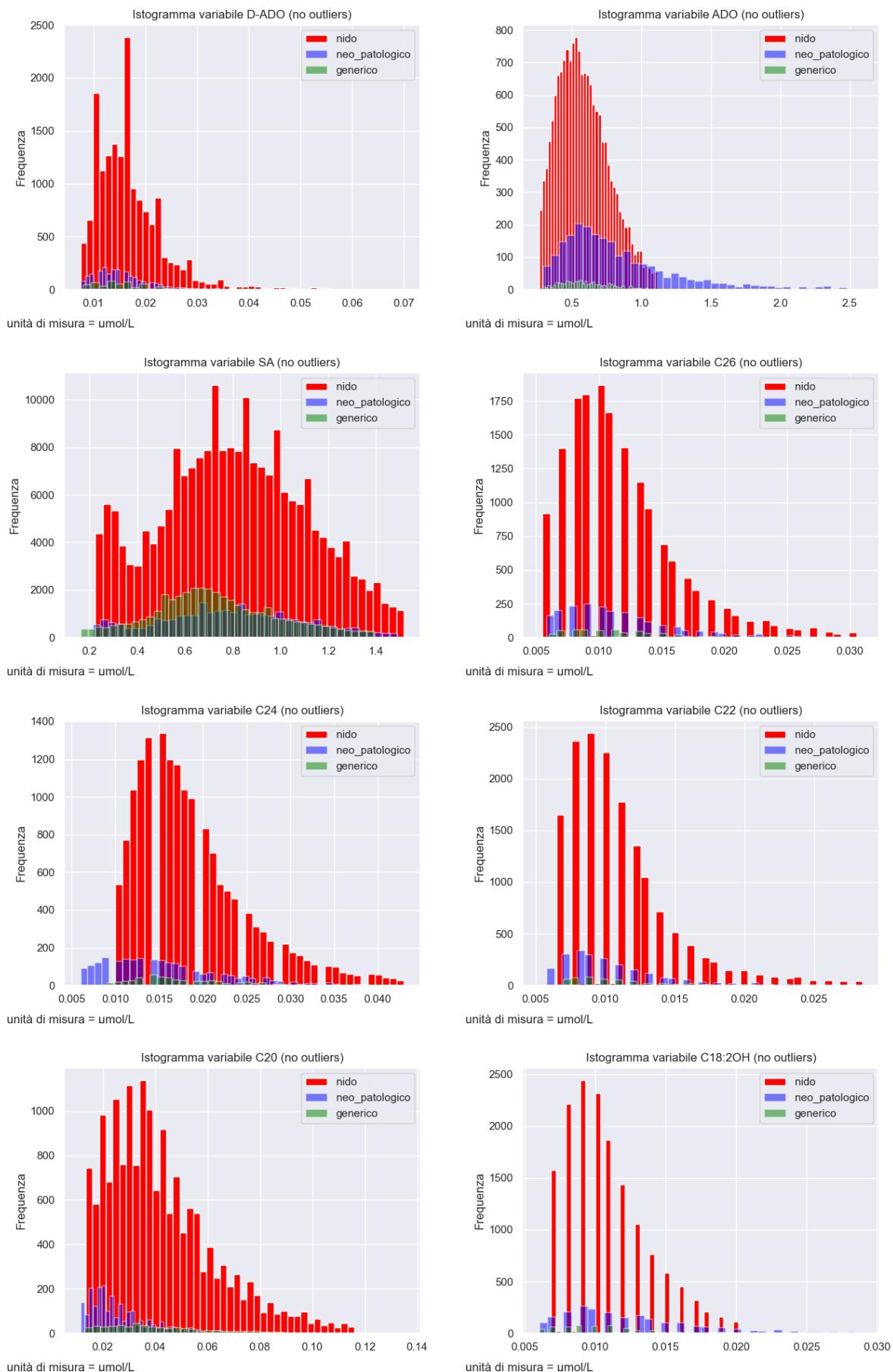
Figure A1.33: istogrammi variabili quantitative (no outliers)

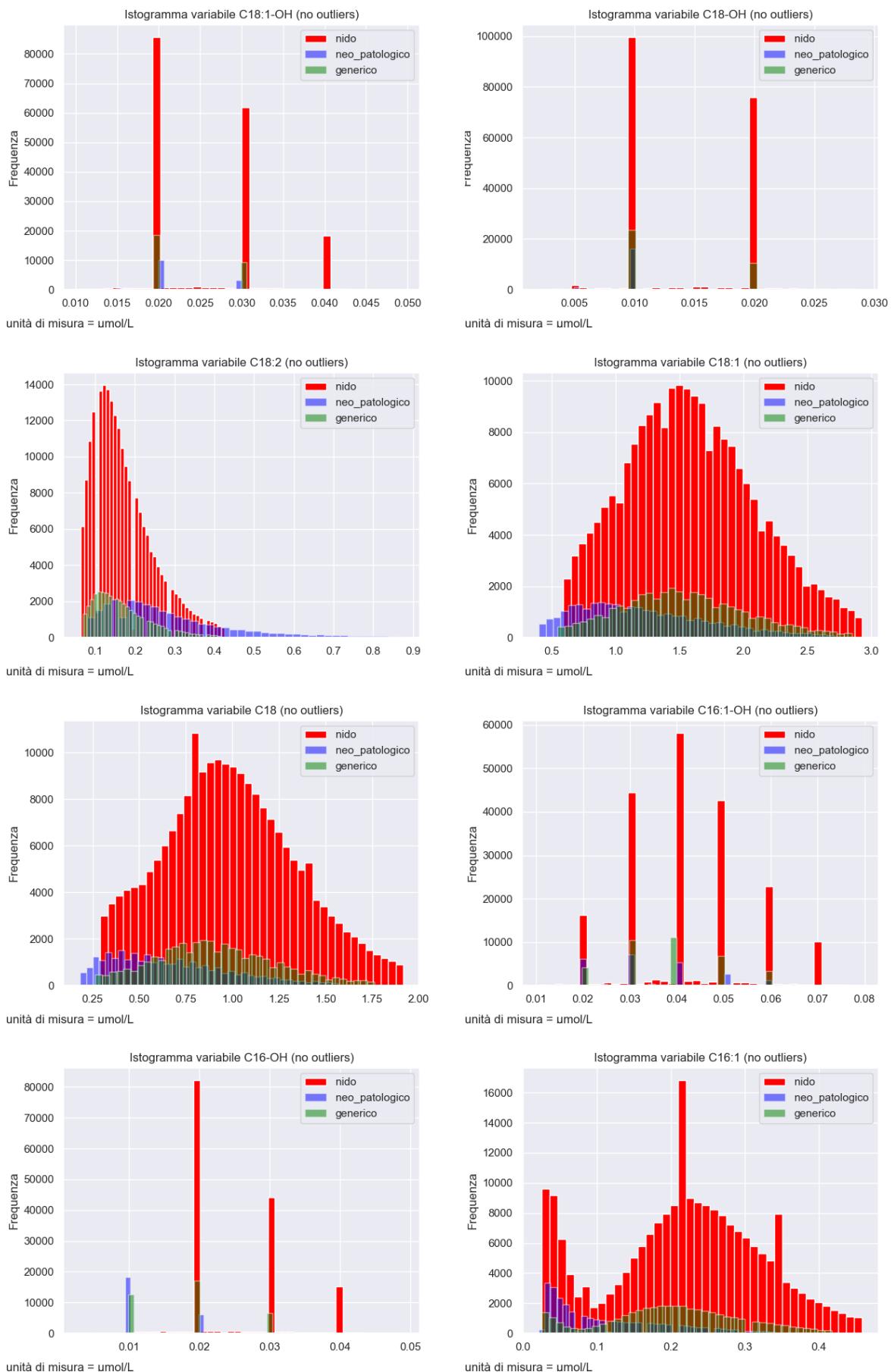
Figure analisi esplorative stratificate per reparto

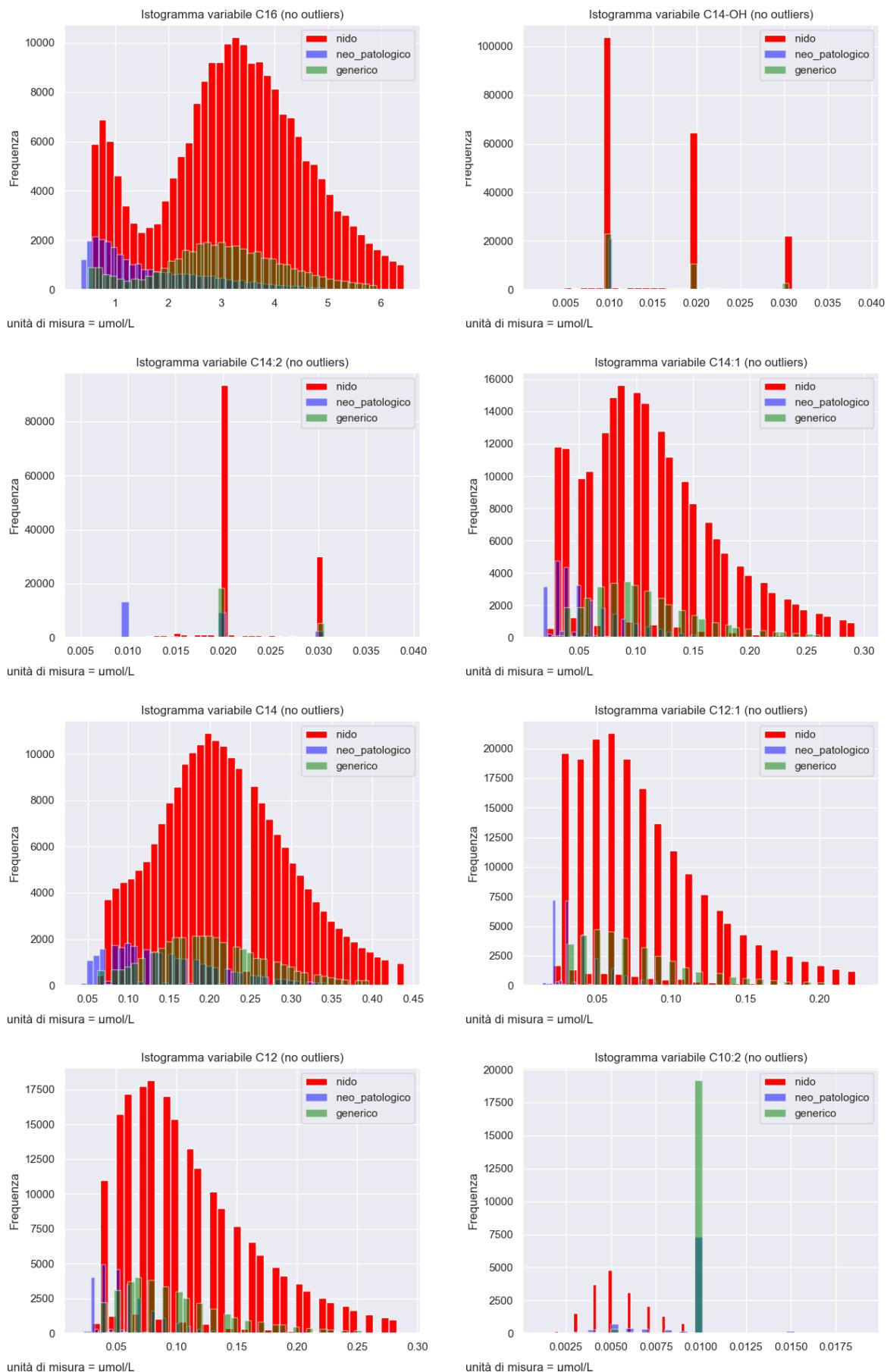


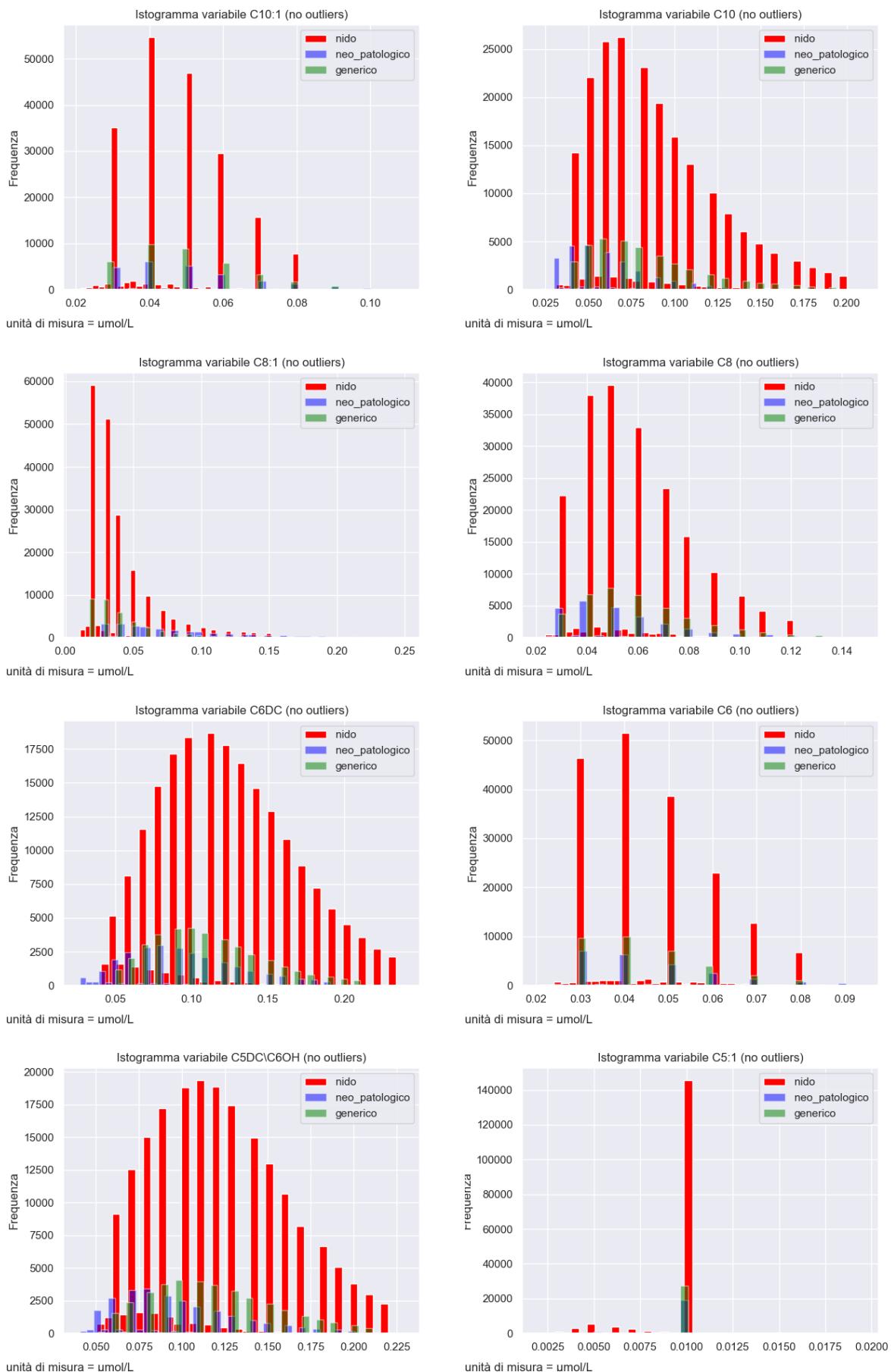


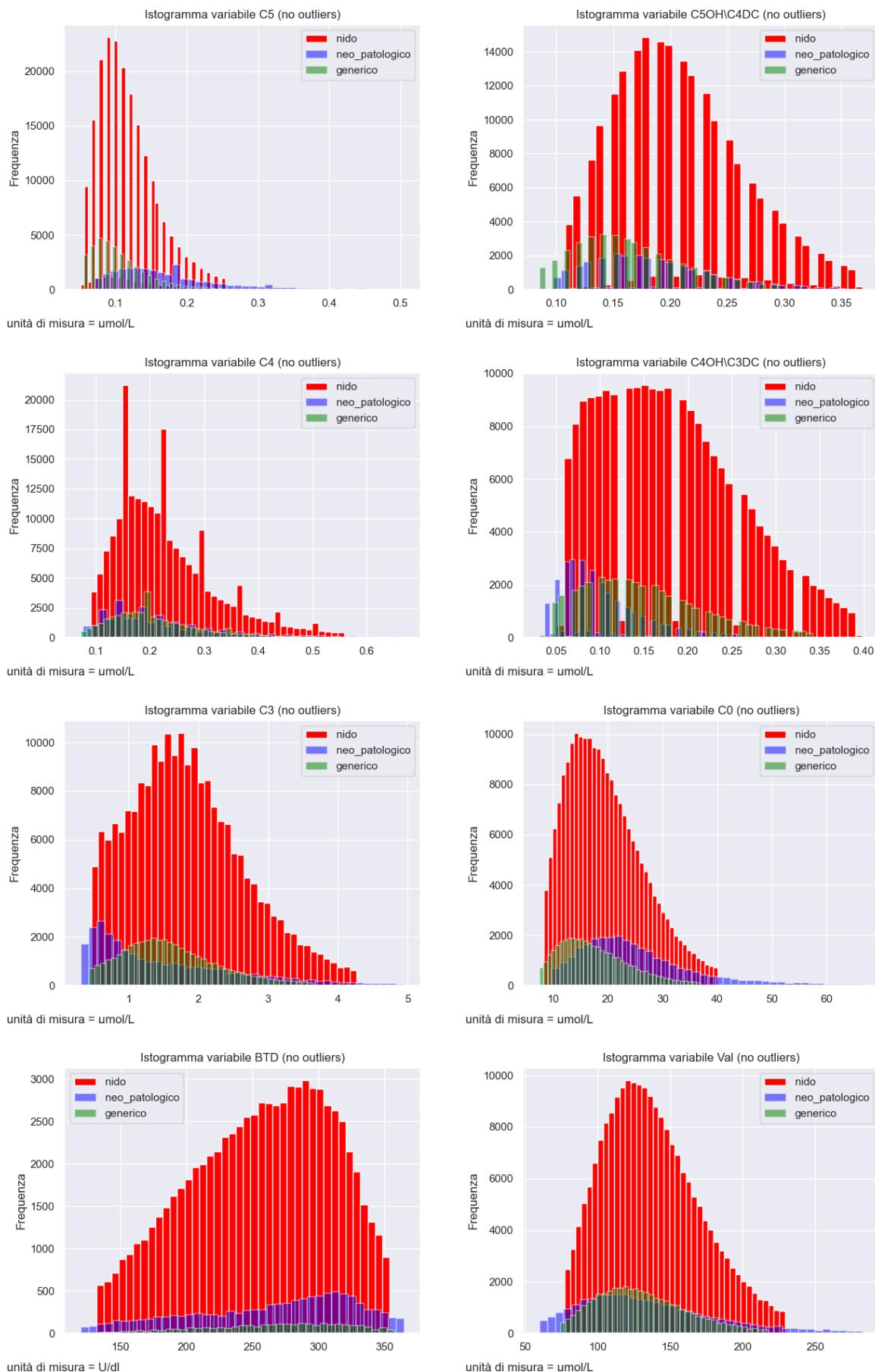


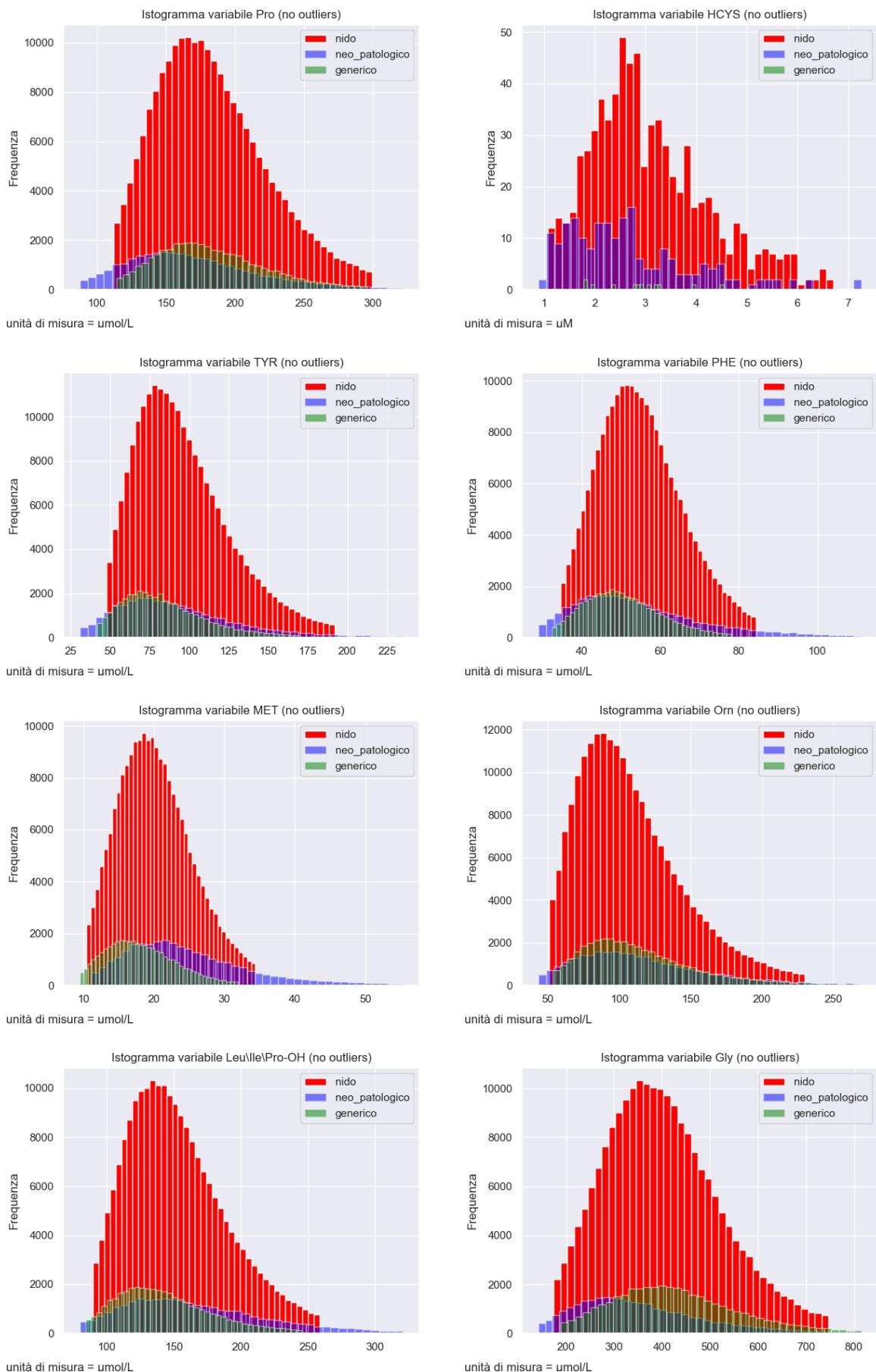












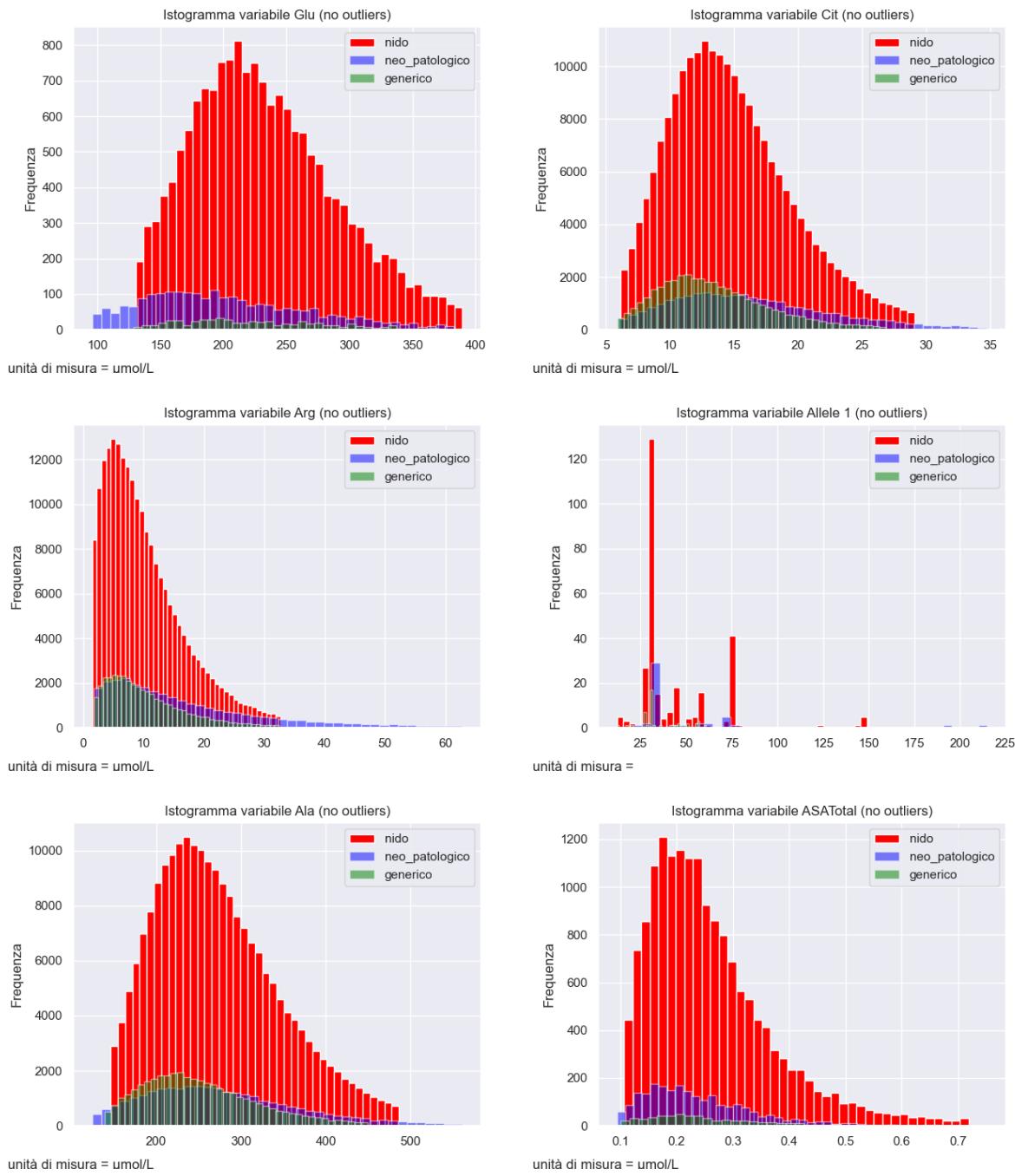
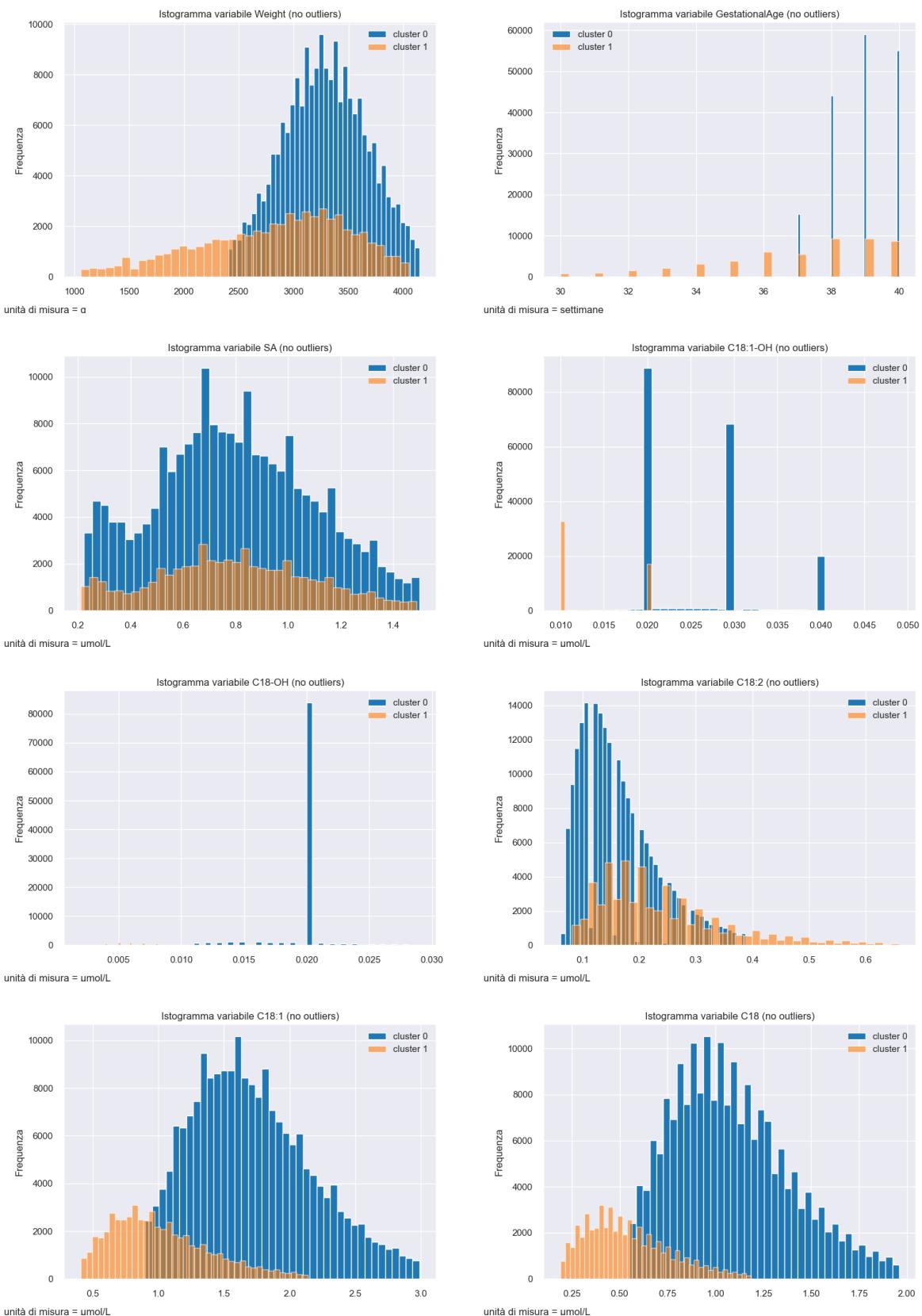
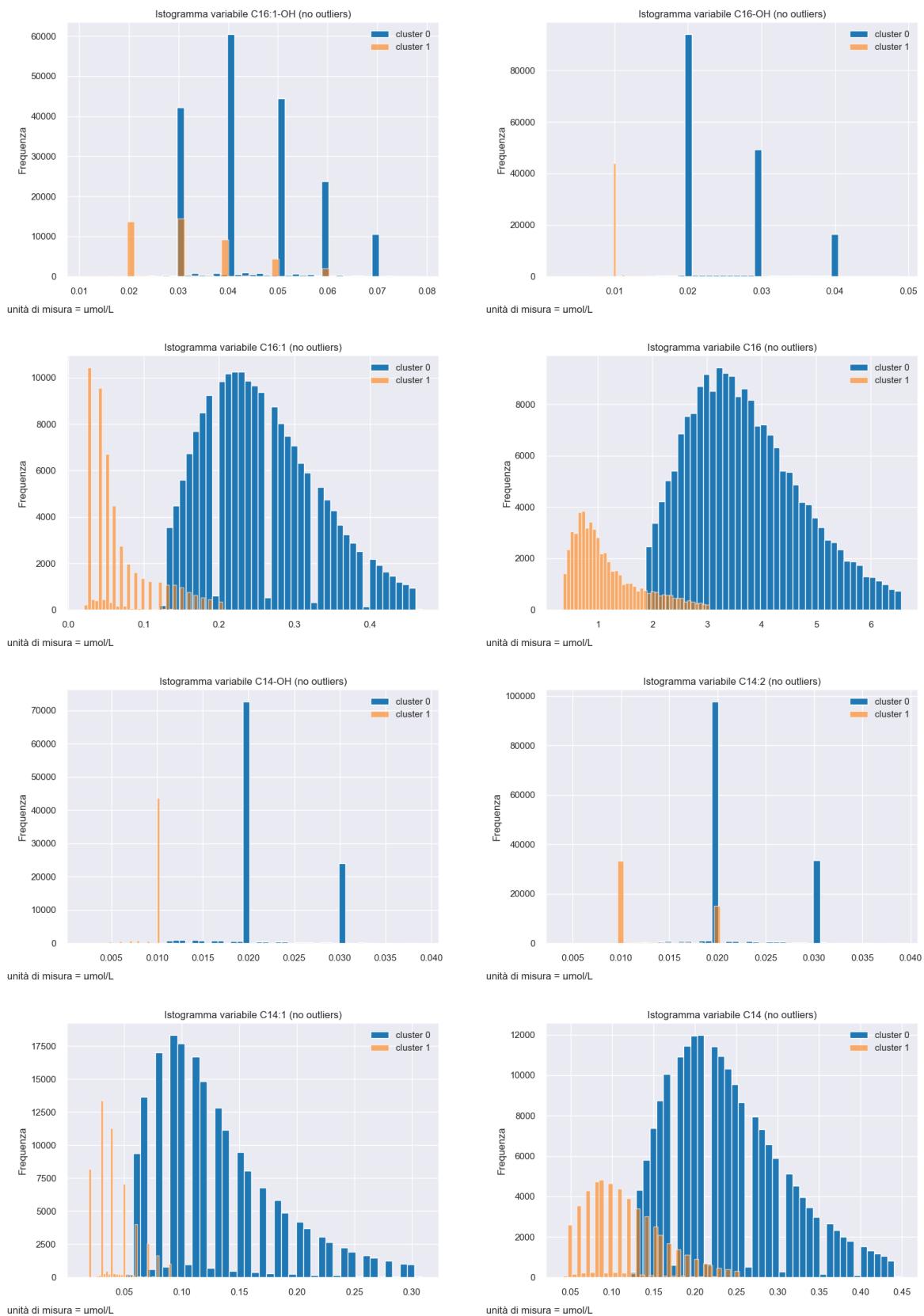
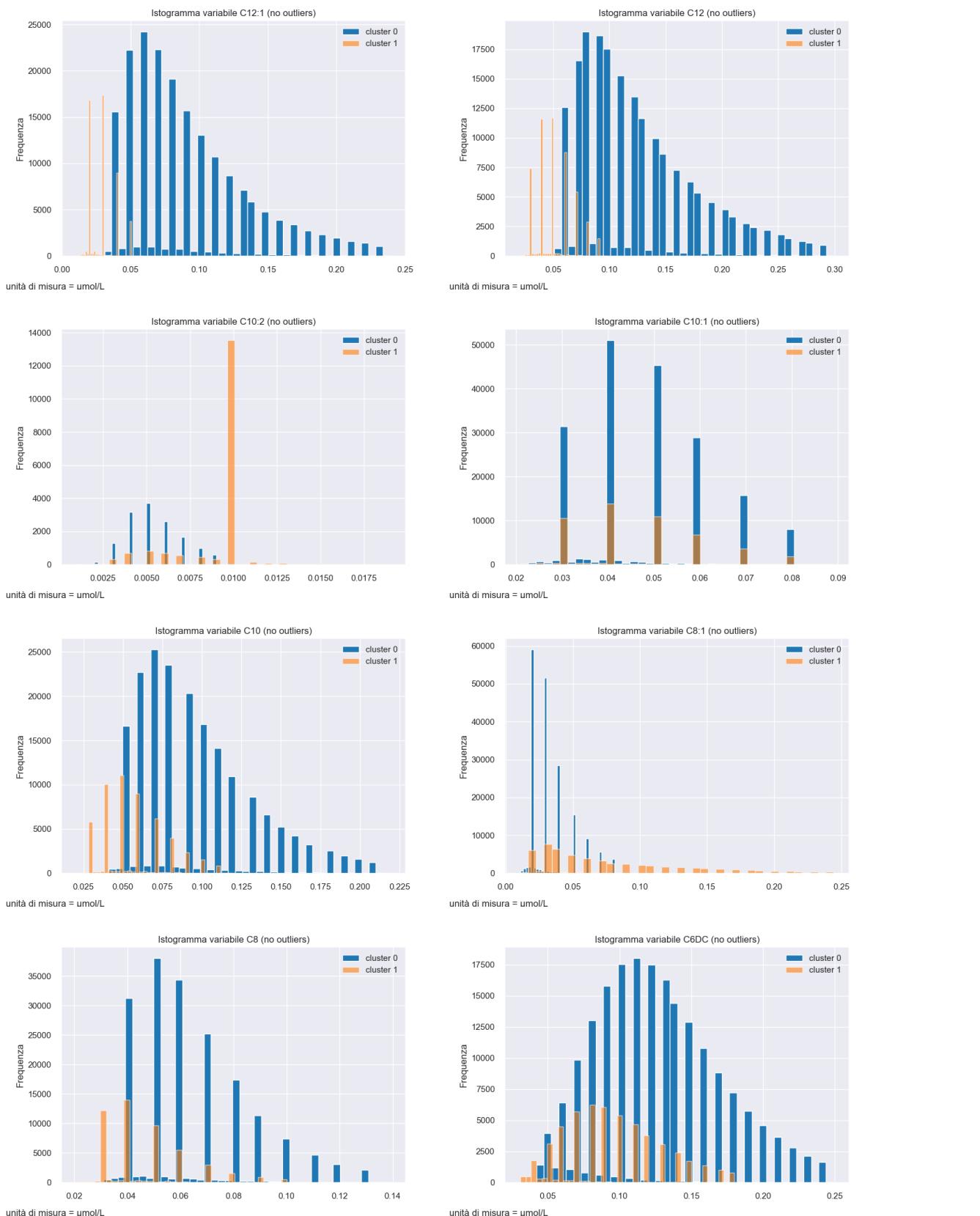


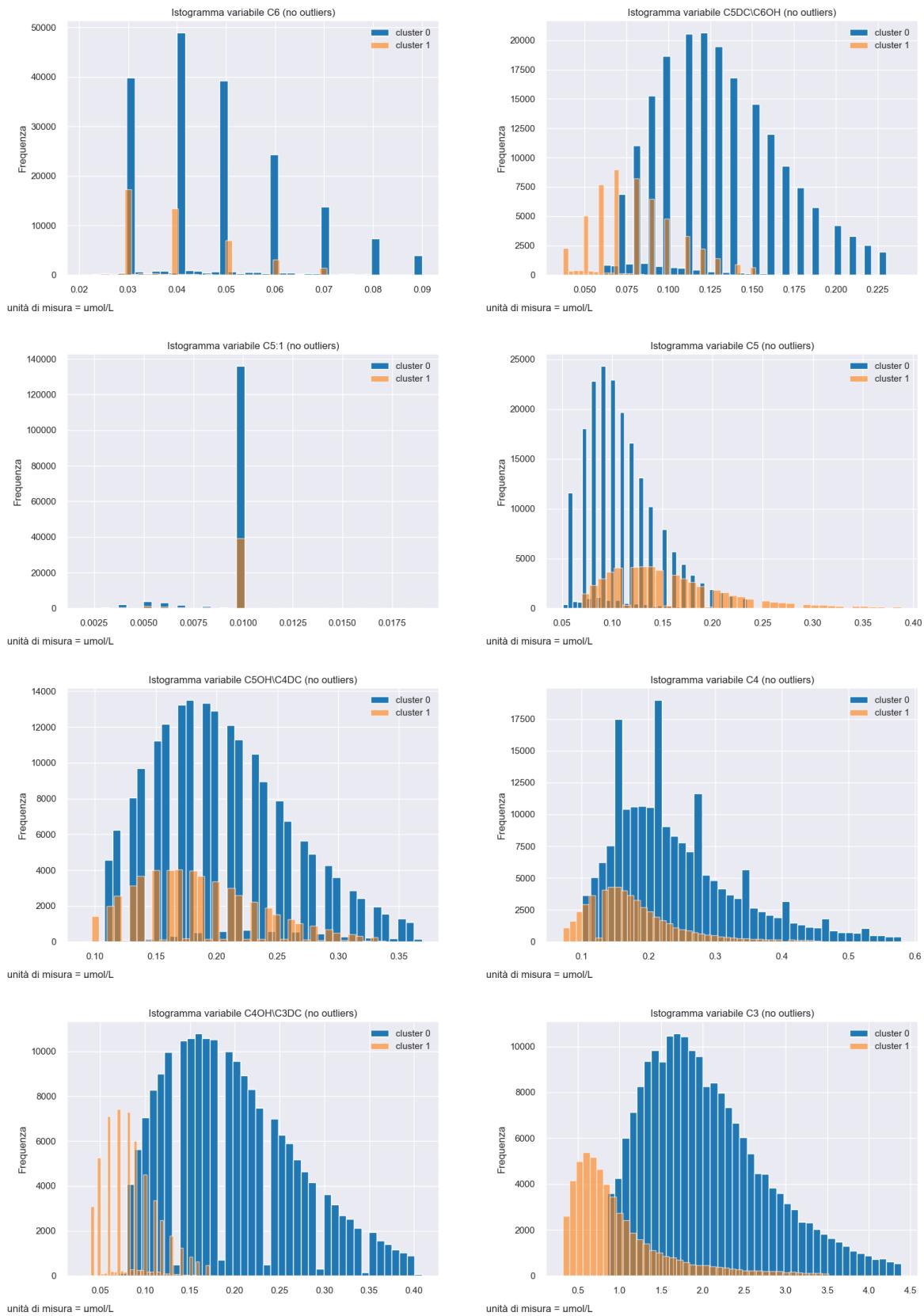
Figure A2.34: istogrammi variabili quantitative (no outliers) stratificate per Reparto

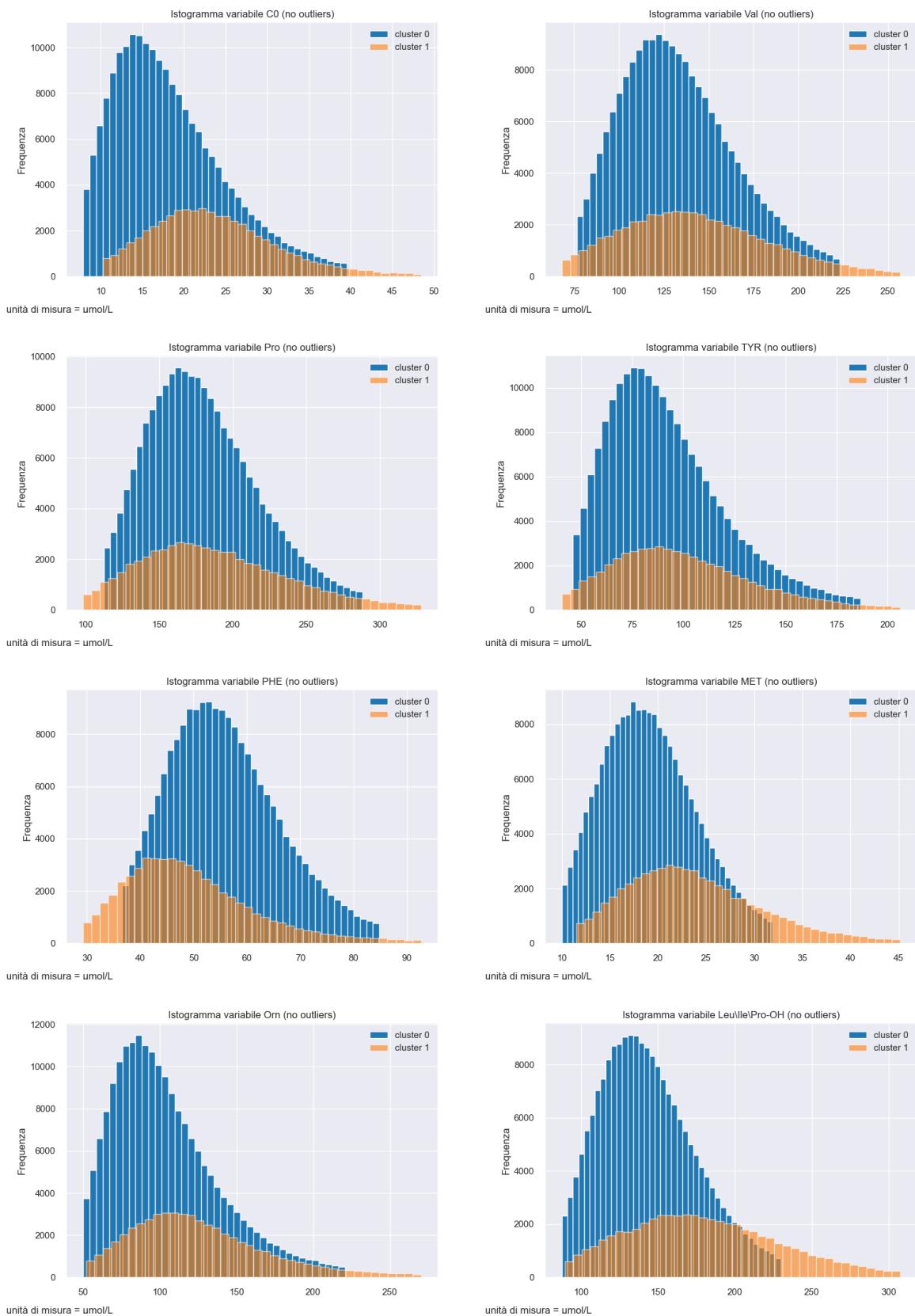
Figure analisi esplorative stratificate per cluster











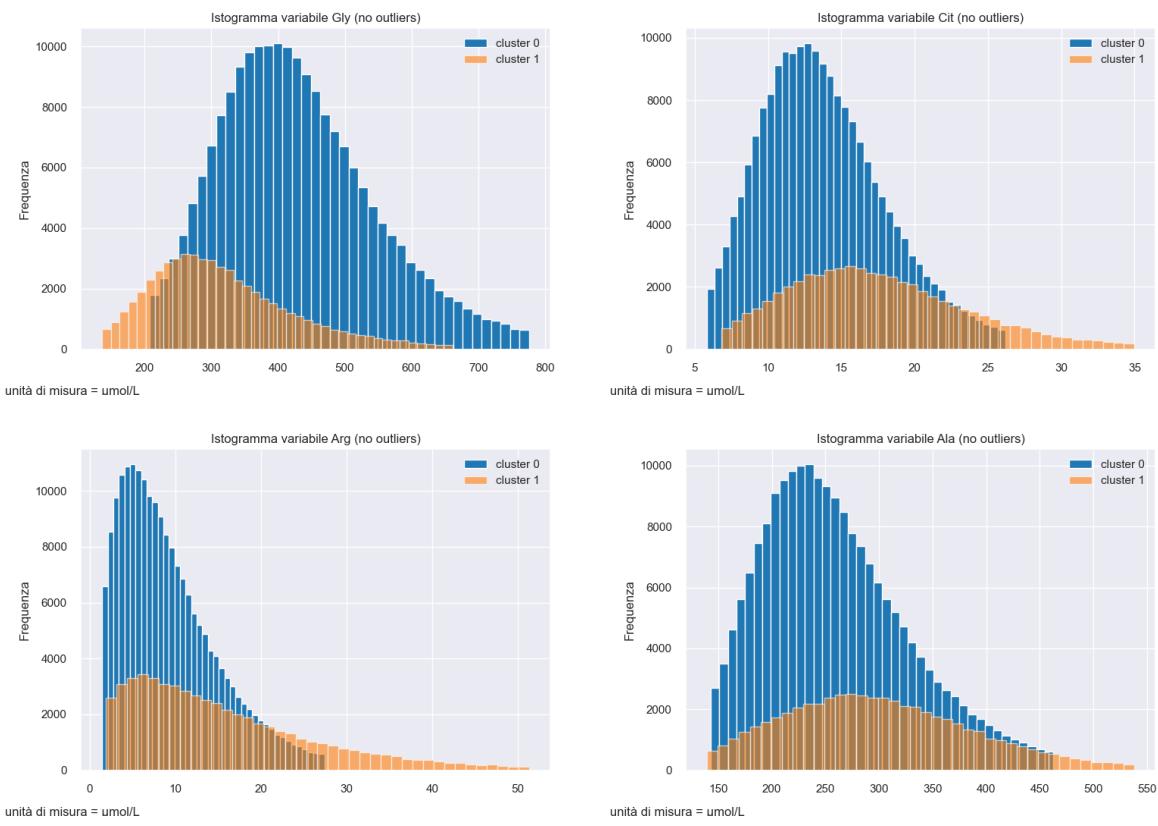


Figure A3.35: istogrammi variabili quantitative (no outliers) stratificate per Cluster

Indice delle figure

Figure 1.1: flowchart procedimento di raccolta dati.....	11
Figure 2.2: heatmap matrice di correlazione completa (no valori).....	25
Figure 2.3: heatmap matrice di correlazione variabili con pochi dati mancanti (no valori).....	26
Figure 2.4: barplot variabili qualitative (escluse City ed Hospital)	38
Figure 2.5: esplorazione variabili qualitative con barplot stratificati	39
Figure 2.6: boxplot variabili Weight e Gestational Age stratificate per altre variabili qualitative	40
Figure 2.7: scatterplot variabili Weight e GestationalAge stratificati per variabile qualitativa	41
Figure 2.8: barplot variabili qualitative stratificate per Reparto	56
Figure 3.9: scatterplot UMAP($n_{neighbors}=5, min_dist=0.05$)	76
Figure 3.10: scatterplot UMAP($n_{neighbors}=5, min_dist=0.01$).....	76
Figure 3.11: scatterplot UMAP($n_{neighbors}=5, min_dist=0.1$)	77
Figure 12: scatterplot UMAP($n_{neighbors}=50, min_dist=0.1$)	77
Figure 3.13: scatterplot UMAP($n_{neighbors}=30, min_dist=0.1$).....	78
Figure 3.14: scatterplot UMAP($n_{neighbors}=15, min_dist=0.1$).....	78
Figure 15: scatterplot PCA($n_{components}=2$).....	79
Figure 3.16: scatterplot tSNE($perplexity = 5, early_exaggeration = 12$)	80
Figure 3.17: scatterplot tSNE($perplexity = 50, early_exaggeration = 12$)	80
Figure 3.18: scatterplot tSNE($perplexity = 30, early_exaggeration = 12$)	81
Figure 3.19: scatterplot tSNE($perplexity = 5, early_exaggeration = 50$)	81
Figure 4.20: esempio di dendrogramma	94
Figure 5.21: scatterplot K-means (2 clusters) con dati ridotti UMAP.....	118
Figure 5.22: scatterplot K-means (3 clusters) con dati ridotti UMAP.....	119
Figure 5.23: scatterplot K-means (4 clusters) con dati ridotti UMAP	120
Figure 5.24: scatterplot K-means (5 clusters) con dati ridotti UMAP	121
Figure 5.25: scatterplot K-means (2 clusters) con dati ridotti PCA.....	122
Figure 5.26: scatterplot K-means (3 clusters) con dati ridotti PCA.....	123
Figure 5.27: scatterplot K-means (4 clusters) con dati ridotti PCA.....	124
Figure 5.28: scatterplot K-means (5 clusters) con dati ridotti PCA.....	125
Figure 5.29: scatterplot K-means (2 clusters) con dati ridotti t-SNE.....	126
Figure 5.30: scatterplot K-means (3 clusters) con dati ridotti t-SNE.....	127
Figure 5.31: scatterplot K-means (4 clusters) con dati ridotti t-SNE.....	128
Figure 5.32: scatterplot K-means (5 clusters) con dati ridotti t-SNE.....	129
Figure A1.33: istogrammi variabili quantitative (no outliers).....	147
Figure A2.34: istogrammi variabili quantitative (no outliers) stratificate per Reparto	157
Figure A3.35: istogrammi variabili quantitative (no outliers) stratificate per Cluster.....	163

Indice delle tabelle

<i>Table 2.1: tabella informazioni e statistiche descrittive variabili quantitative</i>	22
<i>Table 2.2: coppie di variabili quantitative con valori di correlazione (positiva o negativa) > 0.5 </i>	28
<i>Table 2.3: valori elevati VIF variabili quantitative</i>	29
<i>Table 2.4: variabili qualitative del dataset</i>	30
<i>Table 2.5: tabelle distribuzioni variabili qualitative</i>	34
<i>Table 2.6: risultati t-test tra variabili con p-value > 0.05.....</i>	43
<i>Table 2.7: Kolmogorov-Smirnov test tra coppie di variabili con p-value > 0.05.....</i>	45
<i>Table 2.8: tabella informazioni e statistiche descrittive variabili quantitative per Reparto = 'Nido'</i>	47
<i>Table 2.9: tabella informazioni e statistiche descrittive variabili quantitative per Reparto = 'Neo-Patologico'</i>	48
<i>Table 2.10: tabella informazioni e statistiche descrittive variabili quantitative per Reparto = 'Generico'</i>	50
<i>Table 2.11: distribuzioni variabili qualitative stratificate per variabile Reparto</i>	52
<i>Table 4.12: clustering gerarchico agglomerativo, metodo di Ward</i>	109
<i>Table 4.13: clustering gerarchico agglomerativo, metodo del legame medio</i>	110
<i>Table 4.14: clustering gerarchico agglomerativo, metodo del legame completo</i>	111
<i>Table 4.15: clustering gerarchico agglomerativo, metodo del legame singolo</i>	111
<i>Table 4.16: clustering gerarchico agglomerativo, metodo del legame singolo, con metriche diverse dall'euclidea ..</i>	112
<i>Table 4.17: BIRCH clustering</i>	113
<i>Table 4.18: DBSCAN clustering</i>	114
<i>Table 4.19: K-means clustering</i>	116
<i>Table 5.20: indici K-means (2 clusters) con dati ridotti UMAP</i>	119
<i>Table 5.21: indici K-means (3 clusters) con dati ridotti UMAP</i>	120
<i>Table 5.22: indici K-means (4 clusters) con dati ridotti UMAP</i>	120
<i>Table 5.23: indici K-means (5 clusters) con dati ridotti UMAP</i>	121
<i>Table 5.24: indici K-means (2 clusters) con dati ridotti PCA</i>	122
<i>Table 5.25: indici K-means (3 clusters) con dati ridotti PCA</i>	123
<i>Table 5.26: indici K-means (4 clusters) con dati ridotti PCA</i>	124
<i>Table 5.27: indici K-means (5 clusters) con dati ridotti PCA</i>	125
<i>Table 5.28: indici K-means (2 clusters) con dati ridotti t-SNE</i>	126
<i>Table 5.29: indici K-means (3 clusters) con dati ridotti t-SNE</i>	127
<i>Table 5.30: indici K-means (4 clusters) con dati ridotti t-SNE</i>	128
<i>Table 5.31: indici K-means (5 clusters) con dati ridotti t-SNE</i>	129
<i>Table 5.32: tabella informazioni e statistiche descrittive variabili quantitative cluster 0.....</i>	131
<i>Table 5.33: tabella informazioni e statistiche descrittive variabili quantitative cluster 1.....</i>	131
<i>Table 5.34: distribuzione delle variabili qualitative stratificate per cluster.....</i>	133

Bibliografia

- [Banks 04] David Banks, Frederick R. McMorris, Phipps Arabie, and Wolfgang Gaul, *Classification, Clustering, and Data Mining Applications*, Springer Science & Business Media, 2004
- [Vark 04] G. N. Vark, W. W. Howells, *Multivariate Statistical Methods in Physical Anthropology*, D Reidel Pub Co; 1984
- [Lee 07] John A. Lee, Michel Verleysen, *Nonlinear Dimensionality Reduction*, Springer, 2007
- [Nielsen 16] Frank Nielsen, *Introduction to HPC with MPI for Data Science*, Springer, 2016
- [Brito 07] Paula Brito, Guy Cucumel, Patrice Bertrand, Francisco Carvalho, *Selected Contributions in Data Analysis and Classification*, Springer, 2007
- [Cicchitelli 17] G. Cicchitelli, P. D'Urso, M. Minozzo, *Statistica: principi e metodi*, Pearson, 2017
- [Reddy 20] G. Thippa Reddy, M. Praveen Kumar Reddy, Kuruva Lakshmanna, Rajesh Kaluri, Dharmendra Singh Rajput, Gautam Srivastava, Thar Baker, *Analysis of Dimensionality Reduction Techniques on Big Data*, 10.1109/ACCESS.2020.2980942, 2020
- [Maaten 09] L. van der Maaten, E. Postma, J. van den Herik, *Dimensionality Reduction: A Comparative Review*, TiCC TR 2009–005, 2009
- [Ghodsi 06] A. Ghodsi, *Dimensionality Reduction A Short Tutorial*, 2006
- [Murtagh 13] F. Murtagh, M. J. Kurtz, *A History of Cluster Analysis Using the Classification Society's Bibliography Over Four Decades*, arXiv:1209.0125v2 [cs.DL], 2013
- [Kalyani 12] P.Kalyani, *Approaches to Partition Medical Data using Clustering Algorithms*, 2012
- [Manipur 18] I. Manipur, I. Granata, L. Maddalena, M. R. Guaracino, *Clustering analysis of tumor metabolic networks*, 2018
- [Aliusef 22] M. H. Aliusef, G. V. Gnyloskurenko, A. V. Churylina, I. O. Mityuryayeva, *Clustering patterns of metabolic syndrome: A cross-sectional study in children and adolescents in Kyiv*, 2022
- [Kim 16] S. M. Kim, M. I. Pena†, M. Moll, G. Giannakopoulos, G. N. Bennett, L. E. Kavraki, *An Evaluation of Different Clustering Methods and Distance Measures Used for Grouping Metabolic Pathways*, 2016
- [Amigo 08] E. Amigo, J. Gonzalo, J. Artiles, F. Verdejo, *A comparison of extrinsic clustering evaluation metrics based on formal constraints*, 2008
- [Godwin 18] O. Godwin, F. N. Ugwoke, *Clustering algorithm for a Healthcare dataset using Silhouette Score value*, 2018
- [Palacio-Nino 19] J.-O. Palacio-Nino, F. Berzal, *Evaluation Metrics for Unsupervised Learning Algorithms*, 2019
- [Kumar 14] V. Kumar, J. K. Chhabra, D. Kumar, *Performance Evaluation of Distance Metrics in the Clustering Algorithms*, 2014

-
- [Kapil 16] S. Kapil, M. Chawla, *Performance Evaluation of K-means Clustering Algorithm with Various Distance Metrics*, 2016
- [Grabusts 11] P. Grabusts, *The choice of metrics for clustering algorithms*, 2011
- [Karamizadeh 13] S. Karamizadeh, S. M. Abdullah, A. A. Manaf, M. Zamani, A. Hooman, *An Overview of Principal Component Analysis*, 2013
- [Qureshi 17] N. A. Qureshi, H. Magsi, M. J. Sheikh, M. Pathan, *Application of Principal Component Analysis (PCA) to Medical Data*, 2017
- [Kurita 20] T. Kurita, *Principal Component Analysis (PCA)*, 2020
- [Maaten 08] L. van der Maaten, G. Hinton, *Visualizing Data using t-SNE*, 2008
- [Cai 22] T. Tony Cai, Rong Ma, *Theoretical Foundations of t-SNE for Visualizing High-Dimensional Clustered Data*, 2022
- [Cao 17] Y. Cao, L. Wang, *Automatic Selection of t-SNE Perplexity*, 2017
- [Diaz-Papkovich 20] A. Diaz-Papkovich, L. Anderson-Trocmé, S. Gravel, *A review of UMAP in population genetics*, 2020
- [Becht 18] E. Becht, L. McInnes, J. Healy, C.-A. Dutertre, I. W H Kwok, L. Guan Ng, F. Ginhoux, E. W. Newell, *Dimensionality reduction for visualizing single-cell data*, 2018
- [McInnes 20] L. McInnes, J. Healy, J. Melville, *UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction*, 2020
- [Hozumi 21] Y. Hozumi, R. Wang, C. Yin, G.-W. Wei, *UMAP-assisted K-means clustering of large-scale SARS-CoV-2 mutation datasets*, 2021
- [Bridges 66] C. C. Bridges, *Hierarchical cluster analysis*, 1966
- [Papin 21] G. Papin, S. Bailly, C. Dupuis, S. Ruckly, M. Gainnier, L. Argaud, E. Azoulay, C. Adrie, B. Souweine, D. Goldgran-Toledano, G. Marcotte, A. Gros, J. Reignier, B. Mourvillier, J.-M. Forel1, R. Sonneville, A.-S. Dumenil, M. Darmon, M. Garrouste-Orgeas, C. Schwebel, J.-F. Timsit, *Clinical and biological clusters of sepsis patients using hierarchical clustering*, 2021
- [Vaura 20] F. C. Vaura, V. V. Salomaa, I. M. Kantola, R. Kaaja, L. Lahti, T. J. Niiranen, *Unsupervised hierarchical clustering identifies a metabolically challenged subgroup of hypertensive individuals*, 2020
- [Murtagh 12] F. Murtagh, P. Contreras, *Algorithms for hierarchical clustering: an overview*, 2012
- [Zhang 97] T. Zhang, R. Ramakrishnan, M. Livny, *BIRCH: A New Data Clustering Algorithm and Its Applications*, 1997
- [Zhang 96] T. Zhang, R. Ramakrishnan, M. Livny, *BIRCH: An Efficient Data Clustering Method for Very Large Databases*, 1996
- [Ramadhani 20] F. Ramadhani, M. Zarlis, S. Suwilo, *Improve BIRCH algorithm for big data clustering*, 2020

[Lorbeer 17] B. Lorbeer, A. Kosareva, B. Deva, D. Softi, P. Ruppel, A. Küpper, Variations on the Clustering Algorithm BIRCH, 2017

[Khan 14] K. Khan, S. U. Rehman, S. Fong, S. Sarasvady, *DBSCAN: Past, Present and Future*, 2014

[Dubey 17] A. Dubey, A. Choubey, *A Systematic Review on K-means Clustering Techniques*, 2017

[Kumar 18] R. Saravana Kumar, P. Manikandan, *Medical Big Data Classification Using a Combination of Random Forest Classifier and K-means Clustering*, 2018

[Hamerly 03] G. Hamerly, C. Elkan, *Learning the k in K-means*, 2003

[Boutsidis 15] C. Boutsidis, A. Zouzias, M. W. Mahoney, P. Drineas, *Randomized Dimensionality Reduction for K-means Clustering*, 2015

[Ahmed 20] M. Ahmed, R. Seraj, S. M. Shamsul Islam, *The K-means Algorithm: A Comprehensive Survey and Performance Evaluation*, 2020

[distill.pub] https://distill.pub/2016/misread-tsne/?_ga=2.135835192.888864733.1531353600-1779571267.1531353600

[scikit-learn.org] <https://scikit-learn.org/stable/index.html>

[umap-learn.readthedocs.io] <https://umap-learn.readthedocs.io/en/latest/>

[towardsdatascience.com] <https://towardsdatascience.com/>

[github.io] <https://jakevdp.github.io/PythonDataScienceHandbook/05.09-principal-component-analysis.html>

[alessiovaccaro.com] <https://www.alessiovaccaro.com/index.php>

[learndatasci.com] <https://www.learndatasci.com/>

[neptune.ai] <https://neptune.ai/>

[datacamp.com] <https://www.datacamp.com/>

[r-project.it] <http://www.r-project.it/>

[analyticsvidhya.com] <https://www.analyticsvidhya.com/blog/>

[w3schools.com] <https://www.w3schools.com/>

[matplotlib.org] <https://matplotlib.org/>

[pypi.org] <https://pypi.org/project/matplotlib/>

[geeksforgeeks.org] <https://www.geeksforgeeks.org/>

[seaborn.pydata.org] <https://seaborn.pydata.org/>

[pandas.pydata.org] <https://pandas.pydata.org/docs/index.html>

[python-graph-gallery.com] <https://python-graph-gallery.com/>

[pypi.org 2] <https://pypi.org/project/tableone/>

Ringraziamenti

Ringrazio tutta la mia famiglia, per il grande sostegno sempre mostrato, per la fiducia che hanno sempre riposto in me e nelle mie capacità, per la spinta che mi hanno dato nei momenti di difficoltà e per il bene che ci vogliamo ogni giorno, sempre intatto nonostante la distanza.

Ringrazio i miei amici, compagni di una vita e di momenti leggeri ma significativi, sempre pronti a strapparmi una risata nel momento del bisogno.

Ringrazio il dott. Luca Marconi, per l'enorme sostegno e disponibilità in questi mesi di lavoro, per l'entusiasmo, la pazienza e lo spirito con cui abbiamo affrontato insieme i problemi e le sfide di questo progetto, e per i profondi valori umani sempre dimostrati.

Ringrazio il prof. Federico Cabitza, per la grande fiducia riposta nei miei confronti, per i consigli puntuali e gli spunti sempre molto significativi in un progetto così ampio e complesso.

