Modele și tehnici computaționale selectate pentru analiza datelor corelate cu rezistența antimicrobiană

efectuat în cadrul proiectului *Abordarea bioeconomică a agenților antimicrobieni – utilizare și rezistență*

(cod - PN-III-P1-1.2-PCCDI-2017-0361).

Colectiv de redacţie:

Raluca Mureşan, Claudia Zaharia, Radu Moleriu: metode de analiză a asocierilor (sect. 2)

Kristian Miok: modele de învăţare automată în procesarea datelor medicale (sect. 3)

Daniela Zaharie: introducere, metode de clasificare cu etichete multiple (sect. 1)

Coordonator: Daniela Zaharie

Membri: Claudia Zaharia, Raluca Mureşan, Radu Moleriu, Kristian Miok

Data finalizării: 15.11.2019

Raport: R.5.1.2.

Versiunea: 1.1

Acknowledgements

Activities under this work were carried out in the *Research Laboratory Complex "Horia Cernescu"* - financed by project *"A bio-economical approach of the antimicrobial agents - use and resistance"*, in the frame of contract PCCDI 7/19.03.2018, code: PN-III P1-1.2-FPRD-2017.

1. Introducere

În general, seturile de date referitoare la bacteriile cu rezistenţă antimicrobiană pot fi analizate cu modele matematice continue (difuzie în vecinătăţile sferice ale celulelor), modele ordinale (nivele izolate care au o concentraţie minimă de inhibitori) sau modele dihotomice (de exemplu, folosind doar două categorii: bacterii susceptibile sau rezistente). Analiza asocierii în cazul bacteriilor cu rezistenţă antimicrobiană este dificilă datorită faptului că setul de date nu verifică ipotezele modelelor clasice din statistică. Modelele continue şi ordinale nu au o distribuţie normală, iar modelele dihotomice pot conţine date rare şi dispersate. Pe de altă parte, datele colectate se pot caracteriza prin absenţa unor valori ceea ce face analiza şi mai dificilă.

In acest raport este prezentat contextul în care s-a desfăşurat studiul dar şi principalele rezultate obţinute referitor la:

- Identificarea unor asocieri între tipuri de rezistenţă a bacteriilor la diferite antibiotice şi analiza unor măsuri adecvate de ierarhizare a regulilor de asociere extrase din date. Analiza s-a realizat pe un set de date referitor la Escherichia Coli iar rezultatele obţinute sunt în concordanţă cu cele obţinute folosind reţele de interacţiuni între proteine şi un alt set de date referitor la E. Coli (Cărunta et al., 2019).
- Tratarea valorilor absente folosind ca tehnică de imputare mai multe modele generative de tip auto-encoder (Miok et al., 2019a) şi analiza performanţelor acestora atât în cazul unor date de test tradiţionale cât şi în cazul unor date referitoare la caracteristicile producţiei de lapte (inclusiv numărul de celule somatice, interpretate ca potenţial indicator al infecţie) la Staţiunea de Cercetare-Dezvoltare în Creşterea Bovinelor de la Arad.

Rezultatele prezentate în acest raport au fost publicate în două lucrări (autorii subliniaţi sunt membri ai echipei de proiect):

- Kristian Miok, Dong Nguyen-Doan, Marko Robnik-Sikonja and Daniela Zaharie, Multiple Imputation for Medical Data using Neural Network-based Autoencoders Combined with Monte Carlo dropout, The 7th IEEE International Conference on E-Health and Bioengineering - EHB 2019, Iasi, Romania, November 21-23, 2019.
- Claudia Zaharia, Raluca Muresan, Radu Moleriu, Daniela Zaharie, Analysis of association measures used to discover antimicrobial resistance patterns, The 7th IEEE International Conference on E-Health and Bioengineering - EHB 2019, Iasi, Romania, November 21-23, 2019.

2. Metode de analiză a asocierilor

2.1. Concepte de bază

Analiza de asociere este o tehnică utilă pentru a descoperi relaţii interesante între obiecte în seturi de date de mari dimensiuni. Aceste relaţii se exprimă sub formă de mulţimi de elemente frecvente şi reguli de asociere.

În analiza de asociere, datele sunt reprezentate sub forma unei liste de tranzacţii. Dată o mulţime de obiecte $O = \{o_1, o_2, \dots, o_k\}$, vom numi tranzacţie o submulţime t a lui O de obiecte înregistrate în baza de date ca urmare a unei anumite acţiuni. Notăm $T = \{t_1, t_2, \dots, t_N\}$ mulţimea tuturor tranzacţiilor. O reprezentare uzuală pentru T este sub formă de matrice binară, cu N linii (corespunzătoare tranzacţiilor) şi d coloane (corespunzătoare obiectelor). Elementul de pe poziţia (i, j) din matrice va fi 1 dacă obiectul o_j se regăseşte în tranzacţia t_i, respectiv 0 în caz contrar.

O colecţie de obiecte din O poartă numele de *itemset*. O tranzacţie t conţine un itemset X dacă $X \subseteq t$. Frecvenţa unui itemset se defineşte prin numărul tuturor tranzacţiilor care-l conţin:

$$\sigma(X) = card\{t_i \in T : X \subseteq t_i\}.$$

O regulă de asociere este o implicaţie de forma $A \to B$,unde $A, B \subset O$ şi $A \cap B = \emptyset$. Mulţimile A şi B se numesc antecedentul, respectiv consecventul regulii.

Puterea unei reguli de asociere se măsoară cu ajutorul suportului şi încrederii (confidenţei). Vom spune că regula $A \to B$ (şi echivalent, itemsetul $A \cup B$) are suportul s dacă o proporţie s a tranzacţiilor din T conţin simultan pe A şi B. Regula $A \to B$ are încrederea c dacă B din totalul tranzacţiilor ce îl conţin pe A, proporţia celor ce îl conţin şi pe B este c. Formal,

$$supp(A \to B) = \frac{\sigma(A \cup B)}{card(T)} = P(A \cap B)$$

şi

$$conf(A \to B) = \frac{\sigma(A \cup B)}{\sigma(B)} = P(B|A).$$

Dacă o regulă de asociere are suportul mic, atunci este posibil ca ea să fie o coincidenţă. Apariţiile ei sunt rare, şi deci regula poate fi considerată nesemnificativă în aplicaţii practice. În ceea ce priveşte încrederea, cu cât aceasta este mai mare, cu atât este mai probabil ca o tranzacţie care îl conţine pe A să îl conţină şi pe B.

Dat un set de tranzacţii T, scopul analizei de asociere este de se genera toate regulile de asociere cu suport şi încredere mai mari decât anumite valori prag $minsup$, respectiv $minconf$. Întrucât numărul total de reguli ce pot fi formulate pentru un set de k obiecte este extrem de mare ($3^k - 2^{k+1} + 1$), analiza exhaustivă a acestora este prohibitivă din punct de vedere computaţional, existând tehnici mult mai avantajoase de filtrare a spaţiului de căutare pentru determinarea regulilor ce satisfac cerinţele de suport şi încredere.

In contextul analizei rezistenţei antimicrobiene, în special în contextul rezistenţei multiple, este important de identificat antibiotice sau clase de antibiotice care sunt sunt detectate frecvent împreună în seturile de date care cuprind informaţii privind rezistenţa sau susceptibilitatea diferitelor tulpini de bacterii la diferite antibiotice.

2.2. Extragerea asocierilor din date

Demersul de extragere a asocierilor din date presupune două etape:

- Generarea tuturor mulţimilor de obiecte (itemset-uri) care au suportul mai mare decât un prag specificat *minsup*. Acestea se vor numi itemset-uri frecvente.
- Generarea tuturor regulilor de încredere mai mare ca *minconf* ce implică itemset-urile frecvente găsite la pasul anterior.

2.2.1. Generarea itemset-urilor frecvente

Un set de date ce conţine k obiecte poate genera $2^k - 1$ itemset-uri, excluzând mulţimea vidă. Datorită faptului că în multe aplicaţii practice numărul de obiecte k implicate în tranzacţii este foarte mare, numărul de itemset-uri ce trebuie analizate este potenţial enorm.
Se poate defini o structură de tip latice pentru determinarea tuturor submulţimilor unei mulţimi de itemi, un exemplu fiind prezentat în figura de mai jos pentru mulţimea $\{a, b, c, d, e\}$.

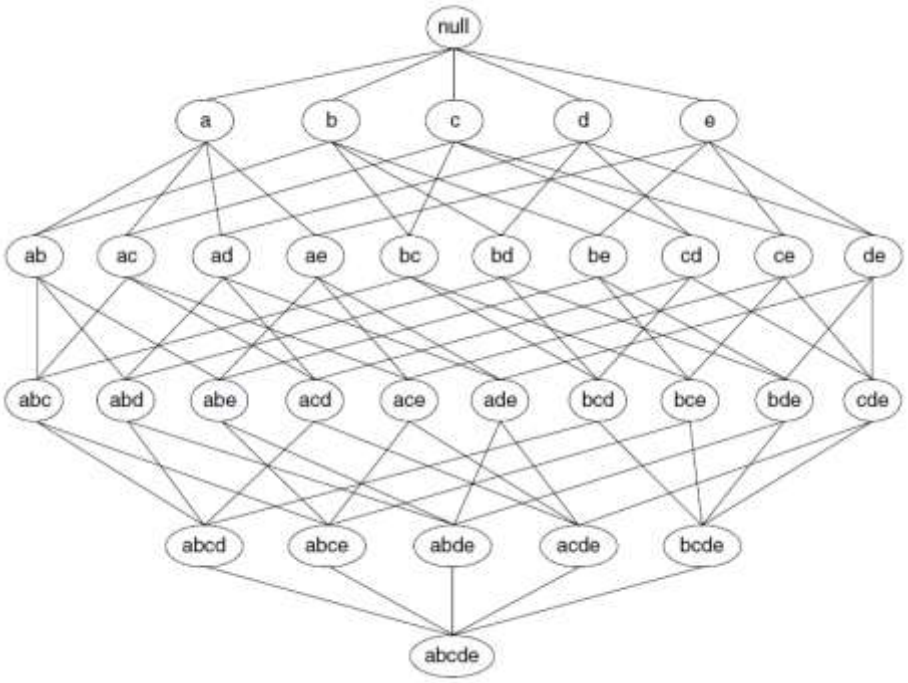

Figura 2.1: Structura de latice a submulţimilor unei mulţimi $\{a, b, c, d, e\}$

O abordare simplistă pentru determinarea itemset-urilor frecvente este calcularea suportului pentru fiecare mulţime candidată din structura latice, comparând-o succesiv cu fiecare tranzacţie. Această metodă însă poate fi foarte costisitoare datorită numărului extrem de mare de comparaţii implicate. O abordare mult mai eficientă foloseşte algoritmul Apriori (Agrawal et al., 1993, Agrawal & Srikant, 1994).

2.2.2. Algoritmul Apriori

Principiul Apriori. Dacă un itemset este frecvent, atunci toate submulţimile sale sunt de asemenea frecvente.

Invers, dacă itemset-ul nu este frecvent, atunci niciunul dintre itemset-urile care îl conţin nu poate fi frecvent. Acest fapt conduce la o strategie numită filtrare bazată pe suport, care foloseşte proprietatea de antimonotonie a măsurii suport, anume că suportul unui itemset nu depăşeşte suportul submulţimilor sale.

Iniţial, fiecare obiect este considerat itemset candidat de dimensiune 1. Dintre acestea, după determinarea suportului fiecăruia, sunt reţinute numai itemset-urile de dimensiune 1 frecvente. La iteraţia următoare, din itemset-urile frecvente de dimensiune 1 sunt alcătuite itemset-uri candidate de dimensiune 2 (niciun intemset frecvent de dimensiune 2 nu poate conţine un itemset ne-frecvent), acestora li se determină suportul, şi sunt păstrate doar itemset-urile de dimensiune 2 frecvente. Similar, la fiecare pas sunt construite itemset-uri candidate de dimensiune k utilizând itemset-uri frecvente de dimensiune $k-1$, apoi dintre acestea sunt reţinute ca frecvente acelea care satisfac restricţia legată de suport. Procedeul se încheie atunci când pentru o anumită dimensiune k nu mai pot fi generate itemset-uri frecvente.

Complexitatea computaţională a algoritmului Apriori depinde de următorii factori:
1. pragul de suport – un prag de suport mai scăzut are ca efect declararea unui număr mai mare de itemset-uri ca frecvente, ceea ce implică necesitatea unui număr mai mare de parcurgeri ale bazei de date.
2. numărul de obiecte din baza de date
3. numărul de tranzacţii
4. numărul mediu de obiecte dintr-o tranzacţie

2.2.3. Generarea regulilor de asociere

Fiecare itemset frecvent Y de dimensiune k poate produce până la $2^k - 2$ reguli de asociere, fără a include regulile care au antecedent sau consecvent vid ($\emptyset \to Y$ sau $Y \to \emptyset$). O regulă de asociere poate fi extrasă prin partiţionarea itemset-ului Y în două submulţimi nevide X şi $Y - X$ astfel încât $X \to Y - X$ să satisfacă pragul de încredere. De menţionat că toate aceste reguli satisfac pragul pentru măsura suport deoarece sunt generate dintr-un itemset frecvent.

Exemplu: Fie $Y = \{a, b, c\}$ un itemset frecvent. Există şase reguli de asociere candidate care pot fi generate din Y: $\{a, b\} \to \{c\}$, $\{a, c\} \to \{b\}$, $\{b, c\} \to \{a\}$, $\{a\} \to \{b, c\}$, $\{b\} \to \{a, c\}$, $\{c\} \to \{a, b\}$. Suportul fiecărei reguli este egal cu suportul mulţimii Y.

Calcularea încrederii pentru fiecare regulă de asociere nu necesită alte evaluări ale mulţimii tranzacţiilor. Considerând regula $\{a, b\} \to \{c\}$, încrederea se calculează ca $suport(\{a, b, c\})/suport(\{a, b\})$. Din moment ce suportul pentru fiecare itemset frecvent a fost deja calculat în timpul generării itemset-urilor frecvente, nu mai este necesar să se parcurgă din nou întregul set de date.

Spre deosebire de măsura suport, încrederea nu are nicio proprietate de monotonie. Totuşi are loc următorul rezultat, valabil pentru compararea regulilor generate din acelaşi itemset frecvent Y:

Dacă o regulă $X \rightarrow Y - X$ nu satisface pragul de încredere, atunci nicio altă regulă $X' \rightarrow Y - X'$, unde X' este o submulţime a lui X, nu va satisface pragul de încredere.

Algoritmul Apriori are o abordare pe nivele pentru generarea regulilor de asociere, unde fiecare nivel corespunde numărului de itemi din consecvent. Iniţial toate regulile cu încredere mare care au doar un item în consecvent sunt extrase. Acestea sunt apoi utilizate pentru a genera alte reguli candidate. De exemplu, dacă $\{a, c, d\} \rightarrow \{b\}$ şi $\{a, b, d\} \rightarrow \{c\}$ sunt reguli cu încredere mare, atunci regula candidată $\{a, d\} \rightarrow \{b, c\}$ este generată prin reunirea mulţimilor consecvente ale celor două reguli. Figura 2.2 de mai jos ilustrează o structură de tip latice pentru regulile de asociere generate din itemset-ul frecvent $\{a, b, c, d\}$. Dacă un anumit nod din latice are o încredere mică, atunci conform teoremei de mai sus întregul subgraf generat de acest nod poate fi eliminat. Dacă, spre exemplu, valoarea încrederii pentru $\{b, c, d\} \rightarrow \{a\}$ este mică, atunci toate regulile ce conţin item-ul a în consecvent, incluzând $\{c, d\} \rightarrow \{a, b\}$, $\{b, d\} \rightarrow \{a, c\}$, $\{b, c\} \rightarrow \{a, d\}$, $\{d\} \rightarrow \{a, b, c\}$ pot fi eliminate.

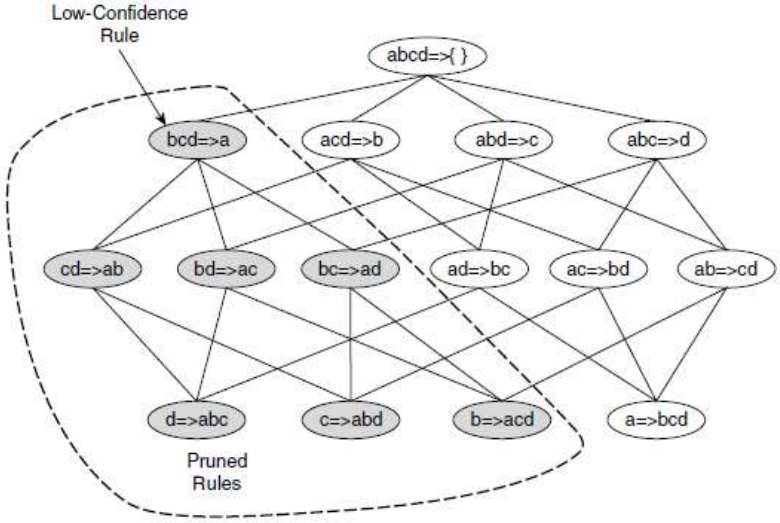

Figura 2.2: Filtrarea regulilor de asociere pe baza încrederii (Tan et al, 2014)

2.3. Măsuri pentru evaluarea calităţii asocierii

O problemă importantă în analiza regulilor de asociere o constituie faptul că, în general, numărul de reguli ce pot fi extrase este foarte mare, aşa încât poate deveni dificil pentru utilizator să le identifice pe cele mai relevante pentru aplicaţia sa. Din acest motiv au fost introduse o serie de măsuri care să ierarhizeze regulile în funcţie de interesul potenţial pe care l-ar putea reprezenta pentru utilizator. Aceste măsuri se împart în trei categorii principale (Geng & Hamilton, 2006):

- *măsuri obiective* - bazate strict pe date, provin din domenii precum teoria probabilităţilor, statistică, teoria informaţiei, învăţare automată;
- *măsuri subiective* – depind de date, dar încorporează şi cunoştinţele sau aşteptările utilizatorului. Astfel de măsuri cuantifică, spre exemplu, gradul de noutate al unei reguli pentru utilizator sau gradul de neaşteptat al regulii, discordanţa faţă de ceea ce este cunoscut sau anticipat în domeniu;
- *măsuri semantice* – au în vedere contextul şi interpretarea regulilor, utilitatea acestora pentru realizarea unui anumit obiectiv.

În cele ce urmează ne vom concentra asupra măsurilor de interes (MI) de tip obiectiv. În literatură există un număr semnificativ de lucrări ştiinţifice dedicate analizei acestui tip de măsuri, a proprietăţilor lor şi a relaţiilor dintre acestea - a se vedea spre exemplu (Piatetsky-Shapiro, 1991), (Bayardo & Agrawal, 1999), (Tan et al., 2004), (Vaillant et al., 2004), (Geng & Hamilton, 2006), (Weiß, 2008) sau (Martínez-Ballesteros et al., 2014).

În general, o MI obiectivă pentru o regulă de tipul $A \to B$ se defineşte cu ajutorul tabelului de contingenţă al regulii:

	B	$\bar{B}$	
A	$n(AB)$	$n(A\bar{B})$	$n(A)$
$\bar{A}$	$n(\bar{A}B)$	$n(\overline{AB})$	$n(\bar{A})$
	$n(B)$	$n(\bar{B})$	N

Tabelul 2.1: Tabelul de contingenţă al regulii $A \to B$

Acest tabel conţine frecvenţele absolute ale tranzacţiilor din baza de date care satisfac anumite condiţii cu privire la elementele din A şi B, respectiv:

- $n(AB)$ reprezintă numărul de tranzacţii care conţin simultan pe A şi B;
- $n(A\bar{B})$ este numărul de tranzacţii care conţin pe A, dar nu şi pe B;
- $n(\bar{A}B)$ este numărul de tranzacţii care conţin pe B, dar nu şi pe A;
- $n(\overline{AB})$ este numărul tranzacţiilor din care lipsesc simultan A şi B;
- $n(A)$ şi $n(\bar{A})$ reprezintă numărul de tranzacţii care conţin, respectiv nu conţin pe A (idem pentru B)
- N este numărul total de tranzacţii din baza de date.

Pe baza acestor valori se definesc probabilităţile corespunzătoare, e.g. $P(A) = \frac{n(A)}{N}$, $P(A \cap B) = \frac{n(AB)}{N}$, $P(B|A) = \frac{P(A \cap B)}{P(A)} = \frac{n(AB)}{n(A)}$.

Primele MI obiective utilizate pentru selectarea şi ierarhizarea regulilor în analiza de asociere au fost suportul şi încrederea, descrise anterior. Alte câteva MI folosite uzual sunt prezentate pe scurt în continuare.

1. *Factorul de interes (liftul)* este definit prin

$$Lift(A \to B) = \frac{P(A \cap B)}{P(A) \cdot P(B)}.$$

Acesta cuantifică în ce măsură apar mai frecvent în tranzacţii A şi B decât ar fi de aşteptat sub ipoteza de independenţă statistică. Este cunoscut faptul că, dacă A şi B sunt independente, atunci $P(A \cap B) = P(A) \cdot P(B)$, prin urmare o valoare mai mare ca 1 a liftului indică o asociere pozitivă între A şi B.

2. *Coeficientul de corelaţie* ϕ este dat de

$$\phi(A \to B) = \frac{P(A \cap B) - P(A) \cdot P(B)}{\sqrt{P(A)P(B)(1 - P(A))(1 - P(B))}}.$$

Acesta provine din statistică şi este analogul coeficientului de corelaţie Pearson folosit în studiul asocierii variabilelor continue. El are valori între -1 (asociere negativă perfectă) şi 1 (asociere pozitivă perfectă), valoarea 0 indicând independenţa statistică între A şi B.

3. *Măsura cosinus* se defineşte ca

$$\cos(A \to B) = \frac{P(A \cap B)}{\sqrt{P(A)P(B)}}$$

şi reprezintă cosinusul unghiului între vectorii de incidenţă a lui A, respectiv B, în tranzacţiile din baza de date. O valoare apropiată de 0 indică lipsa de asociere, în timp ce valorile apropiate de 1 indică o asociere puternică între A şi B.

4. *Coeficientul Jaccard*

$$\zeta(A \to B) = \frac{P(A \cap B)}{P(A \cup B)}$$

reprezintă proporţia tranzacţiilor care conţin simultan pe A şi B din numărul acelora care conţin cel puţin una dintre A şi B. Domeniul de valori al coeficientului Jaccard este [0,1], unde 0 indică lipsa de asociere, iar 1, asociere pozitivă perfectă.

5. *Raportul de şanse ("odds ratio")*

$$\alpha(A \to B) = \frac{P(A \cap B)P(\bar{A} \cap \bar{B})}{P(\bar{A} \cap B)P(A \cap \bar{B})}$$

este o măsură de asociere provenită din statistică. Interpretarea acestuia este următoarea: dacă B este prezent într-o tranzacţie, şansele să fie prezent şi A sunt $\frac{P(A \cap B)}{P(\bar{A} \cap B)}$, în timp ce dacă B este absent, şansele ca A să fie prezent în tranzacţia respectivă sunt date de $\frac{P(A \cap \bar{B})}{P(\bar{A} \cap \bar{B})}$. Dacă raportul acestor două cantităţi este supraunitar, prezenţa lui B sporeşte şansele ca A să fie prezent, deci A şi B sunt asociate pozitiv. Dacă raportul de şanse este subunitar, asocierea este negativă. În sfârşit, dacă raportul de şanse este 1, A şi B nu sunt asociate.

Coeficienţii Y şi Q ai lui Yule sunt variante normalizate ale raportului de şanse, definite respectiv prin

$$Q = \frac{\alpha - 1}{\alpha + 1}, Y = \frac{\sqrt{\alpha} - 1}{\sqrt{\alpha} + 1}.$$

O serie de alte MI obiective folosite în analiza de asociere sunt incluse în Tabelul 2.1 de mai jos. Acestea sunt descrise pe larg în (Hahsler, 2015).

Măsura de interes	**Definiţia**
Coeficientul Goodman-Kruskal (λ)	$\frac{\sum_j \max_k P(A_j \cap B_k) + \sum_j P(A_j \cap B_k) - \max_j P(A_j) - \max_k P(B_k)}{2 - \max_j P(A_j) - \max_k P(B_k)}$
Cohen κ	$\frac{P(A \cap B) + P(\bar{A} \cap \bar{B}) - P(A)P(B) - P(\bar{A})P(\bar{B})}{1 - P(A)P(B) - P(\bar{A})P(\bar{B})}$
Informaţie mutuală	$\frac{\sum_i \sum_j P(A_i \cap B_j) \log \frac{P(A_i \cap B_j)}{P(A_i)P(B_j)}}{\min(-\sum_i P(A_i) \log P(A_i), -\sum_j P(B_j) \log P(B_j))}$

Modele şi tehnici computaţionale selectate pentru analiza datelor

Măsura J	$\max(P(A \cap B)\log\frac{P(B\|A)}{P(B)} + P(A \cap \bar{B})\log\frac{P(\bar{B}\|A)}{P(\bar{B})}, P(A \cap B)\log\frac{P(A\|B)}{P(A)} + P(\bar{A} \cap B)\log\frac{P(\bar{A}\|B)}{P(\bar{A})})$
Indicele Gini	$\max(P(A)[P(B\|A)^2 + P(\bar{B}\|A)^2] + P(\bar{A})[P(B\|\bar{A})^2 + P(\bar{B}\|\bar{A})^2] - P(B)^2 - P(\bar{B})^2, P(B)[P(A\|B)^2 + P(\bar{A}\|B)^2] + P(\bar{B})[P(A\|\bar{B})^2 + P(\bar{A}\|\bar{B})^2] - P(A)^2 - P(\bar{A})^2)$
Laplace	$\max(\frac{NP(A \cap B)+1}{NP(A)+2}, \frac{NP(A \cap B)+1}{NP(B)+2})$
Convingere	$\max(\frac{P(A)P(\bar{B})}{P(A\bar{B})}, \frac{P(B)P(\bar{A})}{P(B\bar{A})})$
Piatetsky-Shapiro	$P(A \cap B) - P(A)P(B)$
Factor de certitudine	$\max(\frac{P(B\|A)-P(B)}{1-P(B)}, \frac{P(A\|B)-P(A)}{1-P(A)})$
Valoare adăugată	$\max(P(B\|A) - P(B), P(A\|B) - P(A))$
Putere colectivă	$\frac{P(A \cap B) + P(\overline{AB})}{P(A)P(B) + P(\bar{A})P(\bar{B})} \cdot \frac{1 - P(A)P(B) - P(\bar{A})P(\bar{B})}{1 - P(A \cap B) - P(\overline{AB})}$
Klosgen	$\sqrt{P(A \cap B)}\max(P(B\|A) - P(B), P(A\|B) - P(A))$

Tabelul 2.2: Măsuri obiective pentru reguli de asociere

În funcţie de proprietăţile sale, o MI poate fi adecvată pentru o anumită clasă de aplicaţii şi nepotrivită pentru altele. Principalele proprietăţi teoretice pe care le poate avea o MI obiectivă m, propuse şi studiate în (Piatetski – Shapiro, 1991) şi (Tan et al., 2004), sunt enumerate în continuare.

(P1) $m(A \to B) = 0$ atunci când A şi B sunt independente
(P2) m creşte cu $P(A \cap B)$ atunci când $P(A)$ şi $P(B)$ rămân constante
(P3) m scade cu $P(A)$ (respectiv $P(B)$) atunci când $P(A \cap B)$ şi $P(B)$ (respectiv $P(A)$) rămân constante
(P4) $m(A \to B) = m(B \to A)$ (simetrie la permutarea variabilelor)
(P5) m este invariantă la scalarea liniilor sau coloanelor tabelului său de contingenţă cu un factor pozitiv
(P6) $m(A \to B) = -m(A \to \bar{B}) = -m(\bar{A} \to B)$
(P7) $m(A \to B) = m(\bar{A} \to \bar{B})$
(P8) valoarea lui $m(A \to B)$ nu este influenţată de tranzacţiile care nu le conţin pe A şi B

Măsura de interes	**P1**	**P2**	**P3**	**P4**	**P5**	**P6**	**P7**	**P8**
Φ	da	da	da	da	nu	da	da	nu
Goodman-Kruskal	da	nu	nu	da	nu	nu *	da	nu
Raport de şanse	da *	da	da	da	da	da *	da	nu
Yule Q	da	da	da	da	da	da	da	nu

Contract 7PCCDI, COD: PN-III-P1-1.2-PCCDI-2017-0361

Yule Y	da	da	da	da	da	da	da	nu
Cohen κ	da	da	da	da	nu	nu	da	nu
Informaţie mutuală	da	da	da	nu	nu	nu *	da	nu
Măsura J	da	nu	nu	nu	nu	nu	nu	nu
Indicele Gini	da	nu	nu	nu	nu	nu *	da	nu
Suport	nu	da	nu	da	nu	nu	nu	nu
Încredere	nu	da	nu	nu	nu	nu	nu	da
Laplace	nu	da	nu	nu	nu	nu	nu	nu
Convingere	nu	da	nu	nu	nu	nu	da	nu
Lift	da *	da	da	da	nu	nu	nu	nu
Cosinus	nu	da	da	da	nu	nu	nu	da
Piatetsky-Shapiro	da	da	da	da	da	da	da	nu
Factor de certitudine	da	da	da	nu	nu	nu	da	nu
Valoare adăugată	da	da	da	nu	nu	nu	nu	nu
Putere colectivă	nu	da	da	da	nu	da *	da	nu
Jaccard	nu	da	da	da	nu	nu	nu	da
Klosgen	da	da	da	nu	nu	nu	nu	nu

da*: da, dacă măsura este normalizată
nu*: simetrie la permutarea de linii sau coloane

Tabelul 2.3: Proprietăţile principalelor măsuri de interes obiective (cf. Tan et al., 2004)

Dat fiind faptul că nu toate MI existente satisfac toate aceste proprietăţi (a se vedea de exemplu Tabelul 2.3 de mai sus), este important de stabilit care dintre (P1) – (P8) sunt relevante în contextul analizei de asociere a rezistenţei antimicrobiene, în care variabilele sunt dihotomice (prezenţă/absenţă a rezistenţei la un anumit antibiotic) şi asimetrice (prezenţa rezistenţei este considerată în investigaţie a fi mai importantă decât absenţa).

Proprietatea (P1) afirmă că o asociere care apare strict aleator ar trebui să aibă interes 0. În practică această cerinţă este prea restrictivă. Ea poate fi relaxată impunând o valoare constantă pentru situaţiile de independenţă (spre exemplu, în cazul liftului şi al raportului de şanse, această valoare este 1).

Proprietatea (P2) spune că, presupunând suporturile lui A şi B fixate, regula va fi cu atât mai interesantă cu cât cele două apar mai frecvent împreună.

Proprietatea (P3) spune că dacă suportul comun al lui A şi B, precum şi cel al uneia dintre acestea, sunt fixate, regula va fi cu atât mai interesantă cu cât cealaltă mulţime are

suport mai mic (i.e., există mai puţine tranzacţii care o conţin doar pe ea). Atât (P2), cât şi (P3) sunt proprietăţi de dorit pentru o MI utilizată în analiza de asociere a rezistenţei la antibiotice.

În ceea ce priveşte (P4), în contextul analizei rezistenţei antimicrobiane, existenţa relaţiei de asociere între A şi B se consideră a fi mai importantă decât direcţia implicaţiei între antecedent şi consecvent, aşadar putem considera că proprietatea nu este esenţială în alegerea unei MI adecvate.

Proprietatea (P5) este utilă în general în studiul asocierilor pentru variabile nominale. Proprietatea (P6) spune că m poate identifica atât asocierile pozitive, cât şi pe cele negative, şi este un caz particular al lui (P7). În (Tan et al., 2004) se arată că măsurile cu proprietatea (P7), numite simetrice, nu sunt potrivite pentru aplicaţii ce necesită un tratament diferenţiat pentru prezenţă vs. absenţă – prin urmare, MI folosite în analiza de asociere a rezistenţei la antibiotice trebuie să fie asimetrice.

Măsurile de interes cu proprietatea (P8) ţin seama doar de tranzacţiile în care se regăsesc cel puţin una dintre A şi B. Aceasta implică faptul că valoarea lui $m(A \rightarrow B)$ nu se modifică atunci când în baza de date se adauga tranzacţii care nu implică pe A sau B (un exemplu de astfel de MI este încrederea). Acest lucru poate fi util când se analizează asocieri între antibiotice la care rezistenţa individuală este foarte rară – astfel de asocieri, deşi potenţial relevante pentru cercetător, nu vor putea fi detectate dacă în baza de date există prea multe probe simultan sensibile la respectivele antibiotice.

2.3.1. Consistenţa diverselor măsuri de interes pentru regulile de asociere

Este de notat faptul că diferite MI pot furniza ierarhizări diferite, uneori contradictorii, ale unui set de reguli, o regulă putând fi considerată interesantă după un anumit criteriu dar nu şi după un altul. Prin urmare, alegerea celor mai potrivite MI este esenţială pentru descoperirea celor mai relevante asocieri într-o aplicaţie dată.

O pereche de MI se consideră ca fiind consistente dacă ele ierarhizează similar un set de reguli. Gradul de asemănare între ordonările regulilor date de două MI poate fi evaluat, spre exemplu, folosind coeficienţii de corelaţie Pearson sau Spearman, măsura cosinus sau inversa normei L2. În (Tan et al., 2004) se arată faptul că metodele enumerate mai sus furnizează rezultate similare în aplicaţii, şi mai mult, dacă valorile rangurilor regulilor din fiecare ordonare sunt unice, toţi aceşti coeficienţi sunt monoton dependenţi unul de celălalt.

2.3.2. Decizii statistice în alegerea valorilor prag utilizate în filtrarea regulilor

În analiza regulilor de asociere trebuie ţinut seama de faptul că nu întotdeauna regulile descoperite reflectă asocieri reale, existând posibilitatea ca unele aşa-zise asocieri să fie descoperiri false sau coincidenţe. Extragerea regulilor de asociere se face pe baza unui set de tranzacţii care reprezintă doar un eşantion din populaţia de interes, aşa încât este de aşteptat ca relaţiile descoperite pe acest eşantion să nu corespundă întocmai celor prezente la nivelul întregii populaţii. Se pune deci problema unei abordări statistice pentru a identifica asocieri reale, care reflectă proprietăţile existente la nivel de populaţie, şi care vor fi valabile şi pentru instanţe viitoare (a se vedea (Hämäläinen & Nykänen, 2008), (Riondato & Vandin, 2014), (Hämäläinen & Webb, 2019)).

Rata descoperirilor false poate fi minimizată prin filtrarea setului de reguli după o anumită MI, alegând un prag convenabil. După cum s-a văzut anterior, filtrarea în algoritmul Apriori se bazează pe măsura suport. În (Tan et al., 2004) se arată faptul că filtrarea regulilor

prin impunerea unui prag inferior de suport are ca şi consecinţă eliminarea cu precădere a asocierilor foarte slabe sau negative. Acesta este un aspect benefic în analiza de asociere privind rezistenţa la antibiotice, întrucât în acest caz de interes sunt asocierile pozitive.

Pe baza consideraţiilor din (Megiddo & Srikant, 1998), în lucrarea (Cazer et al., 2019) se propune o metodologie de simulare pentru estimarea ratei de descoperiri false corespunzătoare unei anumite valori prag a măsurii de interes utilizate în filtrarea regulilor. Pornind de la această idee, în lucrarea (Zaharia et al., 2019) se abordează problema complementară, aceea de a determina pragul inferior pentru o anumită MI astfel încât filtrarea să asigure o rată de descoperiri false mai mică decât o valoare impusă (e.g. 5%).

Procedeul se bazează pe generarea unui număr mare de seturi de date artificiale ("nule") conţinând acelaşi număr de tranzacţii ca şi setul de date original, în care apariţia fiecărui item într-o tranzacţie este simulată ca o variabilă aleatoare Bernoulli de parametru *p*=frecvenţa relativă de apariţie a itemului în setul de date original, independent de ceilalţi itemi. Prin urmare, într-un set de date nul, itemii vor apărea în tranzacţii cu aceeaşi probabilitate ca în datele real observate, dar fără a fi asociaţi între ei. Ulterior pe fiecare set de date se aplică algoritmul Apriori generându-se toate regulile de asociere posibile şi se determină cuantila de ordin 0.95 a seriei de valori a MI. Ideea este aceea că, folosind valoarea cuantilei de ordin 0.95 ca prag inferior în filtrarea după acea MI, cel mult 5% din asocierile rezultate strict aleator (coincidenţe) pot fi descoperite ca reguli. Estimarea pragului inferior de filtrare în setul de date original care asigură rata de descoperiri false dorită (sub 5%) se obţine apoi ca media cuantilelor de ordin 0.95 calculate pentru toate seturile de date nule.

2.4. Studiu de caz în analiza rezistenţei antimicrobiene

Rezultatele prezentate în această secţiune se bazează pe lucrarea (Zaharia et al., 2019).

2.4.1. Descrierea setului de date

Pentru exemplificarea tehnicilor descrise mai sus s-a considerat un set de date conţinând specimene de Escherichia Coli provenite de la subiecţi umani şi informaţii privind susceptibilitatea sau rezistenţa acestora la mai multe antibiotice. Datele au fost extrase din *National Antimicrobial Resistance Monitoring System for Enteric Bacteria (NARMS)* pentru perioada 1996-2016 (NARMS, 2019). Această bază de date a fost înfiinţată în 1996 în Statele Unite ale Americii ca un sistem naţional de monitorizare a rezistenţei antimicrobiene pentru bacterii precum Salmonella enterica, Campylobacter şi Escherichia coli regăsite la subiecţi umani, animale domestice, carne şi produse din carne.

Setul de date extras din NARMS conţine 3583 de specimene, fiecare fiind clasificat ca rezistent sau susceptibil, pe baza pragurilor de concentraţie minimă inhibitoare, la următoarele antibiotice: *ampicilină* (AMP), *amoxicilină – acid clavulanic* (AUG), *ceftriaxonă* (AXO), cloramfenicol (CHL), ciprofloxacină (CIP), acid nalidixic (NAL), gentamicină (GEN), *streptomicină* (STR), *trimetoprim – sulfametoxazol* (COT) şi *tetraciclină* (TET).

Din cele 3583 de specimene considerate iniţial, 1 (0.03%) a fost rezistent la 8 din cele 10 antibiotice, 13 (0.36%) la cel puţin 6 antibiotice, 19 (0.53%) la cel puţin 5, 49 (1.37%) la cel puţin 4, 97 (2.71%) la cel puţin 3, 164 (4.58%) la cel puţin 2 şi 329 (9.18%) la cel puţin un antibiotic. Prevalenţa rezistenţei la fiecare antibiotic poate fi observată în Tabelul 2.3 de mai jos.

	AMP	AUG	AXO	CHL	CIP	NAL	GEN	STR	COT	TET
N	106	14	13	60	10	60	15	147	48	200
%	2.96	0.39	0.36	1.67	0.28	1.67	0.42	4.10	1.34	5.58

Tabelul 2.4: Prevalenţa rezistenţei la cele 10 antibiotice în setul de date iniţial (număr total şi procent)

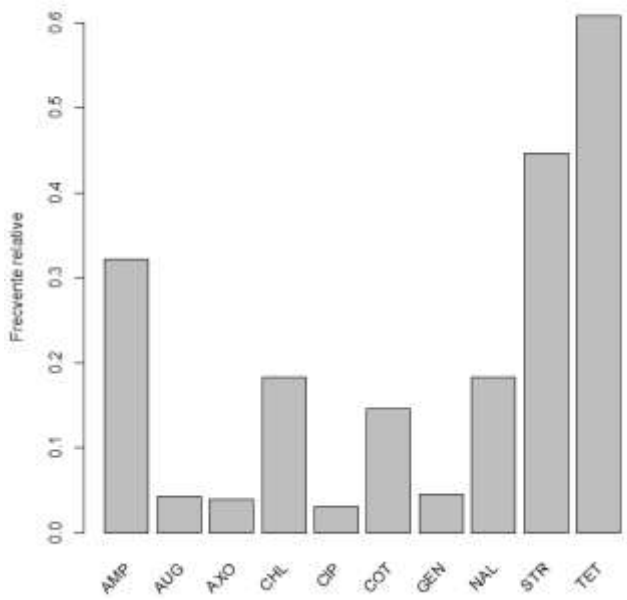

Figura 2.2: Prevalenţa rezistenţei la cele 10 antibiotice în setul de date utilizat în analiza de asociere

Cele 329 de specimene găsite ca fiind rezistente la cel puţin un antibiotic au fost folosite în continuare pentru analiza asocierilor. Extragerea regulilor de asociere s-a realizat utilizând pachetul R "arules" (Hahsler et al., 2019). Pragurile de suport şi încredere au fost stabilite ca $1/(număr\ de\ specimene)$ pentru a genera toate regulile posibile de lungime cel puţin 2. S-au obţinut în total 1329 de reguli de asociere. Acestea au fost evaluate folosind cele 46 de MI implementate în pachetul "arules" (a se vedea (Hahsler et al., 2019) pentru mai multe detalii). O primă selecţie a fost efectuată, eliminând următoarele MI: frecvenţa absolută (coliniară cu suportul, deci redundantă), χ^2 (o variantă scalată a lui Φ^2), şi acele MI care nu au putut fi calculate pentru toate regulile (convingere, raport de şanse, informaţie mutuală, măsura J, Sebag, coeficienţii Yule Q şi Y) sau care au obţinut valoarea ∞ (coeficientul de îmbunătăţire).

A fost evaluată apoi consistenţa MI rămase, calculând coeficientul de corelaţie Pearson pentru vectorii de ranguri ale regulilor generate din setul de date. Distanţa dintre două MI a fost definită ca $1-|r|$, unde r este coeficientul de corelaţie Pearson corespunzător. Distanţele dintre măsuri au avut valori între 0.001 şi 0.999. Se poate observa deci că există perechi de măsuri care ierarhizează similar regulile, cât şi perechi care oferă rezultate discordante. Aplicând, apoi, un algoritm de clustering ierarhic cu metoda "average linkage" au fost identificate cinci grupuri de măsuri cu un comportament similar (a se vedea dendrograma din Figura 2.3).

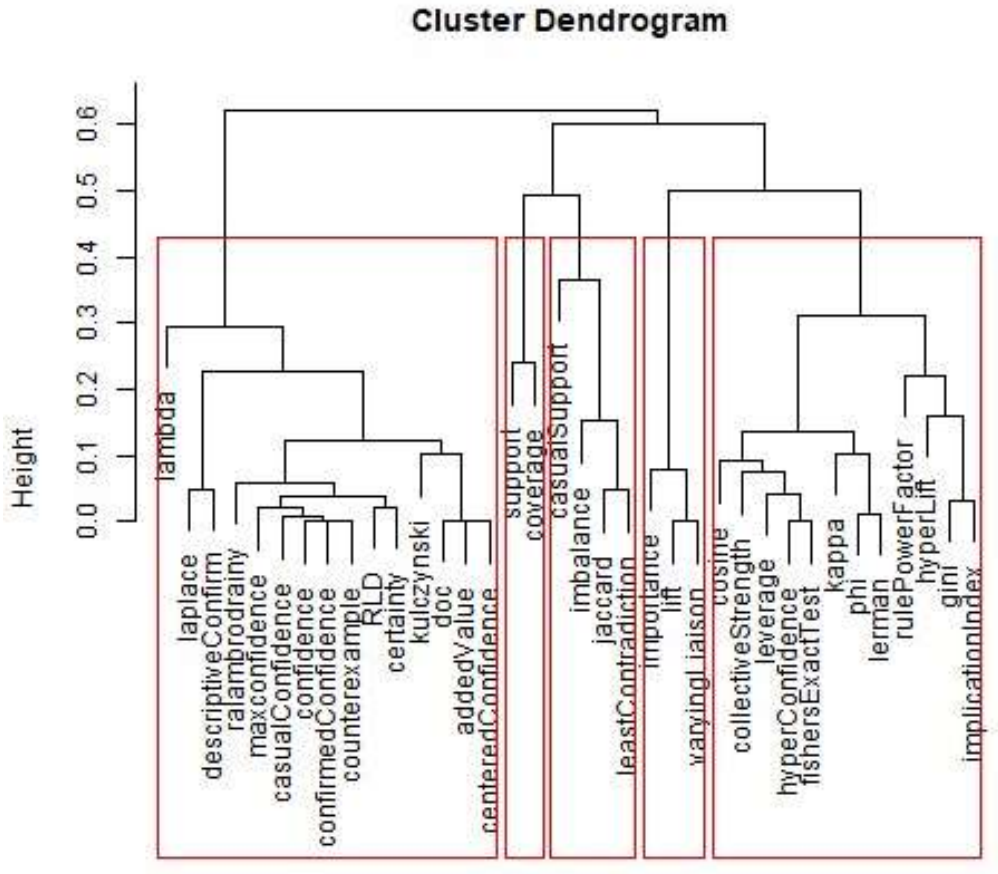

Figura 2.3: Clustering ierarhic al măsurilor de interes

Fiecare cluster conţine măsuri care sunt consistente din punctul de vedere al rangurilor atribuite regulilor extrase din setul de date AMR. Astfel, este suficient să selectăm o măsură reprezentativă din fiecare cluster pentru a determina un clasament agregat al regulilor de asociere.

Conform cu aspectele discutate în secţiunea 2.3, din fiecare cluster a fost selectată o MI reprezentativă, cu proprietăţi adecvate (care să nu satisfacă P7, dar să satisfacă P2 şi P3, asigurându-ne că includem atât MI cu proprietatea P8, cât şi MI care nu satisfac această proprietate). Măsurile selectate sunt: *încredere*, *suport*, *coeficientul Jaccard*, *lift* şi *cosinus*.

În etapa următoare de analiză, a fost determinat pragul minim de suport care să asigure o rată de descoperiri false sub 5%, urmând procedura detaliată în secţiunea 2.3. Au fost considerate 10000 seturi de date nule de câte 329 de tranzacţii, în care fiecare rezistenţă la un anumit antibiotic a fost simulată independent ca variabilă aleatoare Bernoulli de parametru egal cu prevalenţa rezistenţei la antibioticul respectiv în cele 329 tranzacţii iniţiale, şi a fost determinată cuantila de ordin 0.95 a suportului pentru fiecare din cele 10000 seturi de reguli obţinute.

Distribuţia valorilor acestor cuantile ale suportului poate fi vizualizată în Figura 2.4.

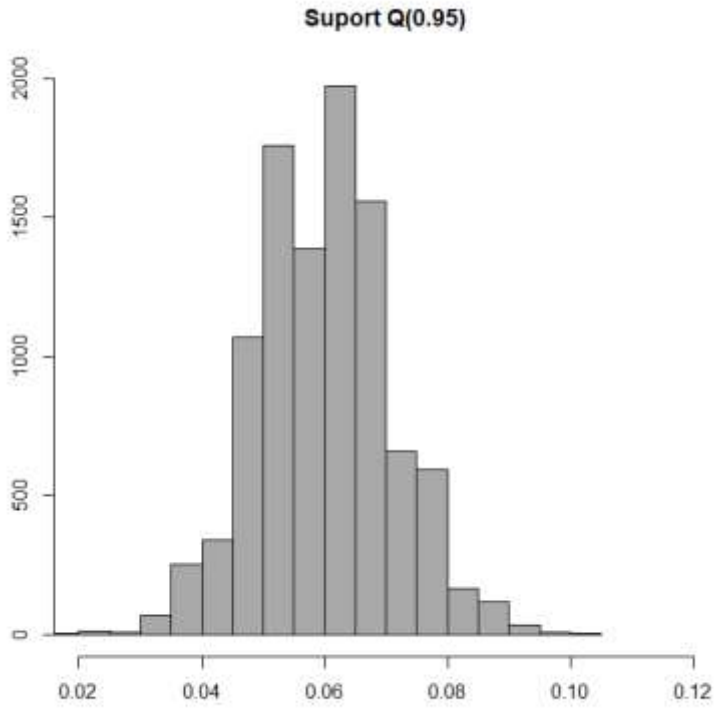

Figura 2.4: Histograma cuantilelor de ordin 0.95 pentru suport în cele 10000 seturi de date simulate

Pragul minim al suportului ce asigură o rată de descoperiri false mai mică decât 5% este 0.06, cu o abatere standard de 0.01. Filtrând setul inițial de reguli folosind această valoare a suportului, numărul de reguli relevante a fost redus la 42. Acestor reguli le-au fost atribuite ranguri folosind fiecare dintre cele cinci măsuri reprezentative menţionate anterior. Pentru fiecare regulă s-a obţinut un rang agregat, calculând media celor cinci ranguri corespunzătoare acestor măsuri.

Regulile ordonate după valoarea rangului agregat (R) sunt prezentate în Tabelul 2.5 de mai jos. Regula cu cel mai mare rang agregat conţine antibioticele CHL, STR şi TET, pentru care cazuri de rezistenţă încrucişată au fost documentate clinic (a se vedea de exemplu (Tadesse et al., 2012)).

A	B	suport	încredere	lift	cosinus	Jaccard	R
{CHL,TET}	{STR}	0.128	0.857	1.918	0.495	0.273	9.9
{STR}	{TET}	0.322	0.721	1.186	0.618	0.440	11
{STR,TET}	{CHL}	0.128	0.396	2.173	0.527	0.339	11.8
{CHL}	{STR}	0.140	0.767	1.716	0.490	0.286	11.9
{AMP,STR}	{COT}	0.082	0.500	3.427	0.530	0.360	12
{TET}	{STR}	0.322	0.530	1.186	0.618	0.440	12.4
{AMP,STR,TET}	{COT}	0.067	0.595	4.075	0.522	0.349	13.3
{AMP,TET}	{COT}	0.076	0.500	3.427	0.510	0.342	13.8
{CHL}	{TET}	0.149	0.817	1.343	0.447	0.232	14.8

{COT}	{STR}	0.116	0.792	1.772	0.452	0.242	15.1
{CHL,STR}	{TET}	0.128	0.913	1.502	0.438	0.206	15.8
{STR}	{CHL}	0.140	0.313	1.716	0.490	0.286	16.5
{COT,TET}	{STR}	0.100	0.846	1.894	0.436	0.216	17
{STR,TET}	{COT}	0.100	0.311	2.134	0.463	0.273	17.1
{COT}	{AMP}	0.094	0.646	2.005	0.435	0.252	17.2
{COT,STR}	{AMP}	0.082	0.711	2.205	0.425	0.231	18.2
{AMP}	{STR}	0.164	0.509	1.140	0.433	0.271	19
{AMP,TET}	{STR}	0.112	0.740	1.656	0.432	0.231	19.2
{STR}	{COT}	0.116	0.259	1.772	0.452	0.242	20.1
{AMP,COT}	{STR}	0.082	0.871	1.949	0.400	0.179	20.2
{AMP}	{COT}	0.094	0.292	2.005	0.435	0.252	20.8
{STR}	{AMP}	0.164	0.367	1.140	0.433	0.271	20.8
{TET}	{CHL}	0.149	0.245	1.343	0.447	0.232	21.2
{COT}	{TET}	0.119	0.813	1.337	0.398	0.187	22.2
{COT,STR}	{TET}	0.100	0.868	1.429	0.379	0.161	23
{COT,TET}	{AMP}	0.076	0.641	1.990	0.389	0.208	23.2
{AMP,COT,TET}	{STR}	0.067	0.880	1.970	0.363	0.147	24.5
{COT}	{CHL}	0.061	0.417	2.285	0.373	0.227	24.6
{COT,STR,TET}	{AMP}	0.067	0.667	2.069	0.372	0.188	24.7
{CHL}	{COT}	0.061	0.333	2.285	0.373	0.227	26.3
{AMP,STR}	{CHL}	0.064	0.389	2.132	0.369	0.226	26.6
{AMP}	{TET}	0.152	0.472	0.776	0.343	0.195	27.5
{AMP,CHL}	{STR}	0.064	0.840	1.880	0.346	0.139	27.8
{TET}	{COT}	0.119	0.195	1.337	0.398	0.187	28.6
{AMP,STR}	{TET}	0.112	0.685	1.127	0.356	0.171	28.8
{STR,TET}	{AMP}	0.112	0.349	1.083	0.349	0.211	30.2
{TET}	{AMP}	0.152	0.250	0.776	0.343	0.195	30.3
{AMP,COT}	{TET}	0.076	0.806	1.327	0.318	0.121	30.6
{AMP,COT,STR}	{TET}	0.067	0.815	1.340	0.299	0.107	31.5
{CHL}	{AMP}	0.076	0.417	1.293	0.313	0.177	33.4
{CHL,STR}	{AMP}	0.064	0.457	1.417	0.301	0.160	34.2
{AMP}	{CHL}	0.076	0.236	1.293	0.313	0.177	35.9

Tabelul 2.5: Regulile finale extrase din setul de date ordonate după rangul agregat

În (Tan et al., 2004), Tan et al. investighează efectul folosirii unui prag inferior de suport asupra distribuţiei coeficientului Φ al regulilor obţinute. Autorii au observat că procedura elimină cel mai adesea reguli cu Φ negativ sau apropiat de 0 (modele necorelate sau slab corelate). În Figura 2.5 se poate observa că fixând un prag minim pentru suport se obţin rezultate similare pentru măsurile Jaccard şi cosinus. Anume, cele mai multe reguli eliminate au valori mici ale celor doi indici, iar cele rămase reprezintă asocieri mai puternice.

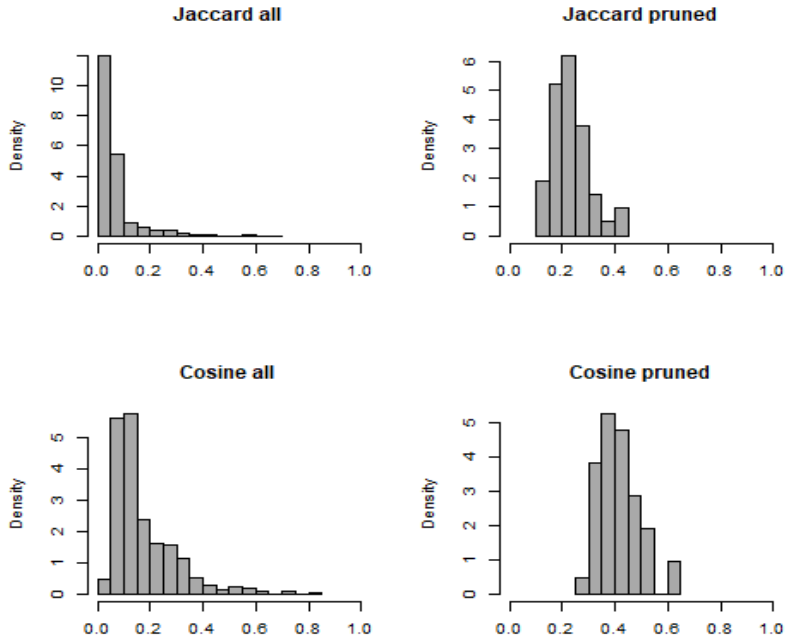

Figura 2.5: Distribuţia măsurilor Jaccard şi cosinus aplicate regulilor înainte şi după filtrarea bazată pe suport

Rezultatele obţinute ilustrează faptul că extragerea regulilor de asociere poate fi un instrument util în determinarea tiparelor de rezistenţă antimicrobiană încrucişată atâta timp cât sunt considerate măsuri de interes relevante.

3. Modele de învăţare automată în procesarea datelor medicale

Rezultatele prezentate în această secţiune se referă la utilizarea unor modele bazate pe învăţare automată în procesarea seturilor de date caracterizate prin valori absente şi se bazează în principal pe lucrările (Miok et al, 2019a), (Miok et al, 2019b).

3.1. Tehnici de imputare a valorilor absente

3.1.1. Problema valorilor absente

Datele incomplete reprezintă una dintre problemele frecvente în analiza datelor şi poate cauza inferenţa din date a unor modele de calitate scăzută şi/sau obţinerea unor concluzii eronate (Lall., 2016). In contextul datelor medicale, prevalenţa valorilor absente poate fi semnificativă din cauza dificultăţii înregistrării datelor (disfuncţionalităţi ale dispozitivelor de măsură, necompletarea unor câmpuri etc.) iar strategia simplă de a exclude înregistrările cu valori absente nu este cea mai adecvată. Tehnicile de tratare a valorilor absente depind de categoria în care se încadrează (Camino et al., 2019):

- Valorile absente sunt complet aleatoare (*missing completely at random* - MCAR) – în acest caz absenţa unei valori poate fi interpretată ca fiind un eveniment aleator

independent atât în raport cu valorile posibile ale atributului pentru care lipseşte valoarea cât şi în raport cu valorile celorlalte atribute.

- Valorile absente sunt aleatoare (*missing at random* - MAR) – în acest caz absenţa unei valori poate fi interpretată ca fiind un eveniment aleator care depinde de valorile celorlalte atribute şi prin urmare probabilitatea ca o valoare să fie absentă poate fi estimată în funcţie de valorile celorlalte atribute.
- Valorile absente nu sunt aleatoare (missing not at random – MNAR) – în acest caz absenţa unei valori depinde atât de valorile posibile ale atributului analizat cât şi de valorile corespunzătoare celorlalte atribute.

3.1.2. Metode de tratare a valorilor absente

Problema valorilor absente poate fi abordată, în funcţie de tiparul specific de absenţă a valorilor, folosind diferite strategii (Van Buuren., 2018):

- *Eliminarea atributului.* În cazul în care pentru unul dintre atribute lipsesc valori corespunzătoare multor instanţe, acel atribut nu conţine suficientă informaţie pentru a fi inclus în analiză, şi ca atare este eliminat.

- *Eliminarea instanţei.* În cazul în care există instanţe pentru care lipsesc valori ale unuia sau mai multor atribute atunci aceste instanţe pot fi excluse din analiză. Este varianta standard de tratare a valorilor absente, fiind varianta implicită în multe pachete de analiză statistică, cum sunt SPSS, SAS şi Stata. Funcţia na.omit() din S-PLUS şi R realizează aceeaşi prelucrare. Avantajele acestei abordări sunt: (i) uşor de implementat; (ii) în cazul în care valorile absente sunt complet aleatoare, prin eliminarea instanţelor cu valori absente nu sunt alterate caracteristicile statistice ale datelor (medie, abatere standard sau estimările coeficienţilor modelelor de regresie). Principalul dezavantaj este faptul că prin eliminarea datelor se pierde informaţie ceea ce conduce creşterea erorilor modelelor statistice construite pornind de la setul redus de date.

- *Eliminare la nivel de perechi*. Ideea acestei metode este de a nu exclude instanţe în întregime ci doar de a nu lua în considerare instanţa dacă analiza implică un atribut pentru care nu este completată valoarea în instanţa respectivă. Spre exemplu în analiza corelaţiei între atribute, pentru fiecare pereche de atribute se vor lua în considerare toate valorile prezente (în felul acesta o instanţă va contribui la valoarea coeficientului de corelaţie în care intervin atributele care au valori completate dar nu va contribui în cazul în care sunt implicate atribute care nu au valori completate). Abordarea aceasta este adecvată doar dacă valorile absente sunt aleatoare.

- *Imputare pe baza mediei.* Tehnicile de imputare au ca scop completarea valorilor absente, pentru a se evita pierderea de informaţie. Cea mai simplă strategie este înlocuirea valorilor absente cu media valorilor existente (în cazul atributelor numerice) respective cu moda valorilor existente (în cazul atributelor categoriale). Principalul avantaj este simplitatea, şi faptul că nu alterează media atributului (în cazul în care valorile absente sunt aleatoare), iar principalul dezavantaj este că va conduce la o sub-estimare a varianţei. Se recomandă utilizarea acestei strategii doar dacă alte strategii nu pot fi aplicate.

- *Imputare pe baza unui model de regresie*. Această tehnică exploatează relaţiile existente între atribute cu scopul de a implementa strategii inteligente de imputare. Primul pas al acestei strategii constă în construirea unui model de regresie pornind de la datele existente. In al doilea pas se calculează, folosind modelul de regresie construit la primul pas, estimări ale valorilor absente. Modelul de regresie poate fi unul simplu sau unul bazat pe tehnici specifice învăţării automate. Principalul avantaj al abordării este că se ţine cont de corelaţiile între atribute, iar principalul dezavantaj este că este o tehnică mai costisitoare decât celelalte. In plus, poate conduce la o sub-estimare a variabilităţii datelor. Din acest motiv este recomandată imputarea multiplă, când unei valori absente i se asociază mai multe valori.

3.1.3. Particularităţile imputării multiple

Tehnicile clasice de imputare generează o singură valoare pentru o observaţie care lipseşte. Dezavantajul acestei abordări este faptul că în procesul de imputare nu se ia în considerare incertitudinea, presupunându-se implicit că valorile generate sunt corecte, adică în procesul de imputare nu intervin erori (ceea ce nu este întotdeauna adevărat). Pentru a elimina acest dezavantaj, pentru fiecare valoare absentă ar putea fi generate mai multe valori, variabilitatea setului de valori putând fi interpretată ca o măsura a incertitudinii.

Această tehnică este denumită *imputare multiplă* (Little & Rubin., 2019). Problema principală în implementarea unei tehnici de imputare multiplă este alegerea adecvată a unui model de imputare (Morris et al., 2014). Unele modele de imputare multiplă nu permit tratarea datelor mixte (categoriale şi continue), altele impun restricţii privind distribuţiile utilizate (de exemplu normalitate) iar altele nu pot gestiona tipare arbitrare de valori absente. Alte dificultăţi pot să apară din cauza dependenţelor/ interacţiunilor neliniare între atribute sau a necesităţii de a procesa un volum mare de date.

3.2. Modele de tip auto-encoders

3.2.1. Invăţare automată şi modele cu structură profundă

Invăţarea automată (Machine Learning) este un domeniu al inteligenţei artificiale având ca scop extragerea de modele pornind de la date. Aceste modele permit descrierea relaţiilor dintre date şi sunt construite pornind de la date folosind algoritmi de învăţare (antrenare). In funcţie de specificul modelului (model de clasificare în clase predefinite sau model de grupare în grupuri de date similare), algoritmii de antrenare pot fi *supervizaţi* (pentru fiecare dintre datele din setul de antrenare se cunoaşte clasa în care se încadrează) sau *nesupervizaţi* (se cunosc doar datele, grupurile nefiind prestabilite). Problemele de clasificare (de exemplu, asocierea unei bacterii cu una sau mai multe familii de rezistenţă antimicrobiană) se încadrează în categoria celor care se rezolvă folosind modele a căror parametri se determină prin algoritmi de antrenare supervizată. Modelele diferă între ele prin structura lor, adică prin modul în care sunt combinate variabilele de intrare şi modul în care intervin parametrii antrenabili. Cele mai simple modele sunt cele liniare, în care variabila de ieşire depinde liniar atât de variabilele de intrare cât şi de parametri.

Modele liniare sunt cu un singur nivel, în sensul că variabila de ieşire poate fi calculată simplu, prin Y=WX, unde X este vectorul cu valorile de intrare iar W este o matrice ce conţine valorile parametrilor antrenabili. Aceste modele au avantajul simplităţii dar dezavantajul că nu pot capta interacţiuni complexe între variabile. Modelele neliniare ierarhice se caracterizează prin faptul că se compun nivele de transformări liniare şi neliniare: $Y=F_n(W_nF_{n-1}(W_{n-1}F_{n-2}(...F_2(W_2F_1(W_1X))...)))$, unde F_i reprezintă o funcţie vectorială neliniară. In această familie de modele intră şi reţelele neuronale cu structură profundă (deep neural networks) care au devenit, în ultimii ani, modele populare în captarea interacţiunilor complexe dintre variabile.

3.2.2. Structura generală a unui auto-encoder

Modelele ierarhice cu structură profundă permit extragerea unor caracteristici ale datelor precum şi ale unor reprezentări mai compacte ale acestora (de exemplu, transformarea unor date caracterizate prin N atribute în date caracterizate prin M<N atribute). Noile atribute pot codifica caracteristici ascunse în datele de intrare şi extragerea lor poate fi benefică în construirea modelelor de procesare ulterioară a datelor (clasificare, regresie sau reconstruire a datelor iniţiale).

Modelele de tip *auto-encoder* (AE) au ca scop transformarea datelor iniţiale în reprezentări mai compacte (care captează cât mai mult din informaţia purtată de datele iniţiale) care ulterior sunt retransformate în date de dimensiunea celor iniţiale, cu scopul de a elimina tot ceea ce este irelevant în datele iniţiale şi a obţine o reprezentare nouă (latentă) a informaţiei stocate în date. Auto-encoder-ele sunt constituite din două componente (Rumelhart et al., 2014):

- un modul de codificare (*encoder*), al cărui scop este de a comprima date de dimensiuni mari în reprezentări de dimensiuni mai mici;
- un modul de decodificare (*decoder*), al cărui scop este să reconstruiască datele iniţiale.

Intrucât în etapa de decodificare, datele sunt reconstruite pornind de la reprezentarea în spaţiul de variabile latente, modelele de tip auto-encoder pot fi utilizate atât pentru a reduce dimensionalitatea datelor (în acest caz se folosesc valorile generate de modulul de codificare) cât şi pentru a genera noi date pornind de la distribuţia datelor iniţiale (în acest caz se foloseşte modulul de decodificare).

In 2013, Kingma şi Welling (Kingma & Welling, 2013) au propus modelele generative de tip variaţional (*Variational Autoencoder* - VAE) care se particularizează în raport cu modelele de tip auto-encoder deterministe prin faptul că în procesul de antrenare se estimează parametrii distribuţiei care generează valorile variabilelor latente (Doersch., 2016). Prin urmare în VAE variabilele latente sunt stocate implicit prin intermediul parametrilor unei distribuţii probabiliste, ceea ce permite generarea unui număr nelimitat de valori având un efect similar procesului de interpolare din cazul determinist.

Arhitectura generală a unui model de tip VAE bazat pe o reţea neuronală multi-nivel este ilustrată în Figura 3.1.

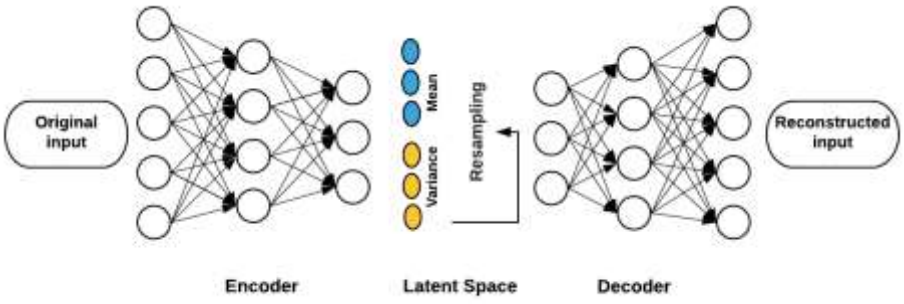

Figura 3.1. Arhitectura unei reţele neuronale generative de tip VAE

3.2.3. Exemple de aplicaţii unde pot fi utilizate modelele de tip auto-encoder

Deşi, iniţial modelele de tip AE şi VAE au fost propuse în contextul prelucrării imaginilor, cu scopul generării de exemple care să extindă seturile de antrenare, la ora actuală sunt utilizate în diferite alte contexte. Printre cele mai populare aplicaţii sunt: reducerea dimensiunii datelor, extragerea caracteristicilor, generarea de noi date. Principiul de bază în utilizarea AE pentru reducerea dimensiunii datelor este că informaţia purtată de datele iniţiale ar putea fi încorporată într-un număr mai mic de variabile, iar aceste variabile sunt cel e obţinute la ieşirea modulului de compresie a unui AE. Principala dificultate este identificarea arhitecturii potrivite, adică a numărului şi dimensiunii nivelelor din structura ierarhică (a reţelei neuronale).

3.2.4. Utilizarea in contextul imputării valorilor absente

Pentru a se realiza imputarea valorilor absente se generează, pornind de la reprezentarea latentă, valori noi folosind tehnica MCMC (Markov Chain Monte Carlo). S-a demonstrat (McCoy et al., 2013) că distribuţia generată folosind MCMC converge către distribuţia marginală a valorilor absente, prin urmare valorile selectate folosind această distribuţie sunt în concordanţă cu distribuţia datelor iniţiale.

Principalele etape ale procesului de imputare bazat pe modele de tip VAE sunt:

- se înlocuiesc valorile absente cu valori stabilite folosind o tehnică simplă de imputare (de exemplu valoarea medie sau moda);
- se transmit datele modulului de codificare din VAE şi se determină parametrii distribuţiei corespunzătoare reprezentării latente a datelor;
- se generează valori folosind distribuţia estimate folosind modulul de codificare
- valorile generate care corespund unor componente absente în datele iniţiale sunt utilizate pentru imputare

Pentru evaluarea calităţii unui model de tip VAE se utilizează ca referinţă un set de date complete şi se elimină din acestea, în manieră aleatoare, o parte dintre valori. Datele de cu valori eliminate sunt utilizate ca date de intrare asupra cărora se aplică etapele enumerate mai sus, iar rezultatul se compară cu datele iniţiale.

In etapa de antrenare, parametrii modelului de tip VAE sunt ajustaţi atât timp cât diferenţa dintre datele ce conţin valori imputate şi datele iniţiale este mai mare decât un prag dat. Procesul de antrenare este influenţat de o serie de parametri de control (hiperparametri): arhitectura şi dimensiunea modelului (numărul de unităţi funcţionale din reţeaua neuronală), parametrii algoritmului de antrenare (algoritmul de antrenare, rata de învăţare, numărul maxim de iteraţii, dimensiunea subsetului de date care constituie un pachet de antrenare, toleranţa de eroare), iar alegerea acestora are impact asupra calităţii modelului, deci implicit asupra calităţii modelului de imputare.

3.3. Instrumente software

Există o serie de instrumente software, majoritatea dezvoltate pentru a fi utilizate din aplicaţii implementate în Python, care permit specificarea şi antrenarea unor modele bazate pe reţele neuronale cu structură adâncă. Unul dintre cele mai populare instrumente este TensorFlow, dezvoltat de Google Brain (https://www.tensorflow.org/) care este o bibliotecă open-source pentru a efectua eficient calculele numerice implicate în modelele specifice învăţării automate, în particular variate arhitecturi de reţele neuronale cu multe nivele. Accesul la funcţiile din TensorFlow este facilitat de biblioteca Keras (https://keras.io/) care oferă o interfaţă intuitivă şi modalităţi simple de descriere a structurii modelelor pornind de la o serie de componente pre-definite.

Fig. 3.2. prezintă un exemplu de descriere în Keras a unui model de tip auto-encoder cu trei nivele ascunse având 2000, 500 respectiv 500 de unităţi funcţionale aplicat pentru imagini de 28*28 pixeli liniarizate ca vectori de 784 de elemente.

```
#defining input placeholder for autoencoder model
input_img = Input(shape=(784,))
# "enc_rep" is the encoded representation of the input
enc_rep = Dense(2000, activation='relu')(input_img)
enc_rep = Dense(500, activation='relu')(enc_rep)
enc_rep = Dense(500, activation='relu')(enc_rep)
enc_rep = Dense(10, activation='sigmoid')(enc_rep)

# "decoded" is the lossy reconstruction of the input from encoded representation
decoded = Dense(500, activation='relu')(enc_rep)
decoded = Dense(500, activation='relu')(decoded)
decoded = Dense(2000, activation='relu')(decoded)
decoded = Dense(784)(decoded)

# this model maps an input to its reconstruction
autoencoder = Model(input_img, decoded)
```

Fig. 3.2. Exemplu de descriere în Keras a unui model de tip auto-encoder

3.4. Studiu de caz în tratarea valorilor absente în seturi de date

3.4.1. Descrierea setului de date

Pentru a ilustra modul în care modelele de tip auto-encoder pot fi utilizate pentru imputarea valorilor absente în cazul datelor bio-medicale au fost utilizate două categorii de date:

- Seturi de date de la UCI (University of California Irvine) repository (Asuncion & Newman, 2007)
- Date colectate la Staţiunea de cercetare-dezvoltare pentru creşterea bovinelor Arad care conţin informaţii privind calitatea laptelui precum şi informaţii relevante pentru analiza rezistenţei antimicrobiene (numărul de celule somatice). Datele au fost colectate de la 264 de vaci şi conţin informaţii cum sunt: cantitate lapte, cazeină, grăsime, lactoză, nivel Ph, proteine, uree şi număr de celule somatice.

Caracteristicile datelor utilizate în analiză sunt ilustrate în Tabelul 3.1.

Table 3.1: Caracteristici ale setului de date: N – număr de instanţe, a – număr de atribute, Num – număr de atribute numerice, Missing – procentul valorilor absente, disc – numărul de atribute discrete, C – numărul de clase.

Datasets	N	a	Num	disc	Missing %	C
Study-MILK	610	11	11	1	3.9	2
Brest-WISC	699	9	9	1	0	2
PIMA-diabetes	768	8	8	1	0	2

3.4.2. Descrierea arhitecturii

Modelul utilizat este cel de tip VAE combinat cu o strategie de Monte Carlo Dropout (MCD) (Gal & Ghahramani, 2016a). Tehnica de tip dropout constă în inactivarea temporară a unor unităţi funcţionale şi a fost aplicată în contextul a diferite arhitecturi de reţele neuronale : feed-forward, convoluţionale, recurente (Gal & Ghahramani, 2015), (Gal & Ghahramani, 2016b). In contextul analizei derulate în cadrul proiectului BioAMR, proprietăţile strategiei VAE+MCD (Helmbold & Philip, 2017) au fost utilizate în scopul estimării valorilor absente. Aplicând tehnica dropout în componenta de decodificare se pot genera mai multe valori la ieşire, iar prin medierea acestora se generează valorile pentru imputare. Combinarea MCD cu modelele de tip AE şi VAE a fost iniţial propusă în (Miok et al., 2019a) cu scopul de a construi modele generative. Avantajul combinării MCD cu VAE este că în etapa de imputare este captată distribuţia specifică datelor şi interacţiunile dintre atribute şi utilizează mecanismul reducerii dimensiunii datelor cu scopul reducerii efectului zgomotului asupra datelor şi asupra aleatorităţii valorilor absente. Arhitectura propusă este descrisă în Fig. 3.3.

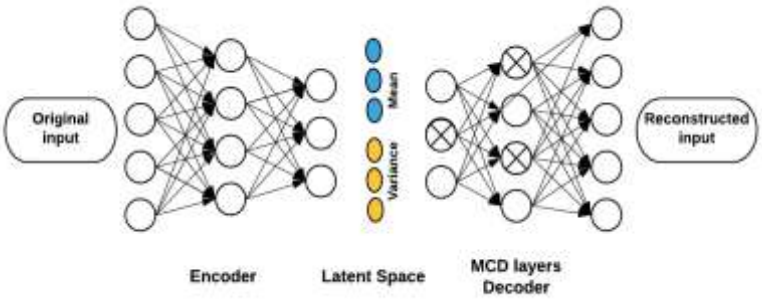

Figure 3.3. Arhitectura modeluluiVAE+MCD.

3.4.3. Metodologia de testare

Pentru a analiza performanţa modelului de imputare, procesul de testare trebuie efectuat într-un mediu controlat, ceea ce înseamnă că se porneşte de la un set de date complete (fără valori absente) din care se elimină în mod aleator, dar controlat, valori pentru a se genera date de test. In acest context, datele de test au fost generate eliminând, în manieră aleatoare, 10%, 30% respectiv 50% din valori (în implementare, valorile selectate au fost inactivate prin înlocuirea lor cu -1). Fiecare set de date este divizat într-un subset de antrenare (90%) şi un subset de testare (10%) utilizat pentru analiza performanţei modelului.

Pentru analiza performanţei modelului sunt folosite două măsuri:

- Eroarea de imputare, măsurată folosind eroarea medie pătratică dintre datele iniţiale (ce conţin valorile reale ale variabilelor absente) şi datele obţinute în urma imputării.
- Acurateţea clasificării (estimată folosind un model de clasificare construit pe baza datelor iniţiale, respectiv pe baza datelor imputate).

Abilitatea de a conserva performanţa unui model de clasificare construit folosind setul complet de date a fost evaluată folosind abordarea propusă în (Robnik-Šikonja., 2019). Ideea de bază este compararea dintre acurateţea modelului construit folosind setul complet de date (dataset1) şi acurateţea modelului construit folosind setul de date cu valori imputate (dataset2): Δacc = accuracy(dataset1) – accuracy(dataset2). O valoare mică a diferenţei sugerează că prin imputarea valorilor absente performanţa predictivă a setului de date poate fi conservată.

3.4.4. **Rezultate**

Arhitectura modelului conţine două nivele ascunse constituite din 80 respectiv 20 unităţi funcţionale. Rata de dropout utilizată este 0.2 (o valoare utilizată frecvent în aplicaţii) iar numărul de epoci de antrenare a fost setat la 300. Intrucât folosind MCD se generează mai multe valori prezise rezultatul este stabilit folosind media valorilor generate. Implementarea este accesibilă la https://github.com/KristianMiok/MI_MCD_VAE.

Rezultatele obţinute folosind validare încrucişată cu 5 felii sunt prezentate în Tabelele 3.2, 3.3. şi 3.4 (Miok et al., 2019c). Cazurile corespunzătoare celei mai mici valori (eroare sau variaţie în acurateţe) sunt marcate.

Table 3.2: Eroarea de imputare (RMSE) pentru cazul în care procentul valorilor absente este 10%: valoare medie (abatere standard).

10% valori absente (RMSE)			
Model	**MILK**	**WISC**	**PIMA**
AE	0.05364[0.0011]	0.07649[0.0093]	0.06565[0.0059]
VAE	0.05027[0.0015]	0.06014[0.0038]	0.06909[0.0083]
MCD-AE	0.0479 [0.0015]	0.06048[0.0053]	0.06649[0.0082]
MCD-VAE	**0.0465 [0.0012]**	**0.05939[0.0029]**	**0.06462[0.0088]**

Table 3.3: Eroarea de imputare (RMSE) pentru cazul în care procentul valorilor absente este 30%: valoare medie (abatere standard)

30% valori absente (RMSE)			
Model	**MILK**	**WISC**	**PIMA**
AE	0.09755[0.0080]	0.12444[0.0170]	**0.11103[0.0046]**
VAE	0.08049[0.0049]	0.11229[0.0091]	0.11666[0.0054]
MCD-AE	0.08685[0.0049]	0.1129 [0.0105]	0.11410[0.0048]
MCD-VAE	**0.07827[0.0051]**	**0.1059 [0.0080]**	0.11221[0.0051]

Table 3.4: Eroarea de imputare (RMSE) pentru cazul în care procentul valorilor absente este 50% valoare medie (abatere standard)

50% valori absente (RMSE)			
Model	**MILK**	**WISC**	**PIMA**
AE	0.12491[0.0104]	0.14901[0.0216]	0.14132[0.0108]
VAE	0.10002[0.0050]	0.13753[0.0092]	0.14057[0.0072]
MCD-AE	0.10559[0.0052]	**0.12488[0.0091]**	0.13829[0.0074]
MCD-VAE	**0.09764 [0.0053]**	0.12706[0.0128]	**0.13815[0.0072]**

Valorile raportate în tabelele 3.2-2.4 sugerează că eroarea de imputare este redusă când se utilizează MCD pentru toate seturile de date şi toate procentele de valori absente, cu excepţia setului PIMA (cu 30% valori absente).

Table 3.5: Compararea proprietăţilor predictive ale modelelor construite folosind datele iniţiale respectiv datele imputate (cazul a 10% valori absente).

Model	MILK	WISC	PIMA
AE	-0.0131	0.01428	0.1365
VAE	-0.0172	-0.00714	0.1429
MCD-AE	**-0.009**	**0**	**0.1230**
MCD-VAE	-0.010	**0**	0.1492

Rezultatele arată ca variantele de imputare bazate pe modele care includ MCD au o mai bună abilitate în conservarea proprietăţilor predictive decât cele care nu includ dropout.

3.4.5. Discuţie

Valorile absente reprezintă o provocare în analiza datelor şi există diferite strategii de imputare, fiecare dintre ele cu avantaje şi dezavantaje. In contextul proiectului BioAMR a fost investigată performanţa unor tehnici de imputare bazate pe modele de tip auto-encoder combinate cu strategii de inactivare a unor unităţi funcţionale (dropout) cu scopul de a realiza imputare multiplă. Rezultatele obţinute indică superioritatea modelelor care încorporează dropout, atât din perspectiva erorii de imputare cât şi din perspectiva proprietăţilor predictive ale setului de date completat cu valorile obţinute în urma imputării. Principalul dezavantaj al abordării propuse este acela că presupune că valorile absente sunt complet aleatoare, ceea ce nu se întâmplă întotdeauna în seturile de date reale.

Bibliografie:

1. (Agrawal et al., 1993) R. Agrawal, T. Imielinski, A. Swami, Mining association rules between sets of items in large databases, in: Proceedings of 1993 ACM-SIGMOD International Conference on Management of Data, Washington, DC, May 1993, pp. 207-216.
2. (Agrawal & Srikant, 1994) R. Agrawal, R. Srikant, Fast algorithms for mining association rules, in: Proceedings of the 20th International Conference on Very Large Databases, Santiago, Chile, 1994, pp. 487-499.
3. (Asuncion & Newman, 2007) Asuncion, Arthur, and David Newman. "UCI machine learning repository." (2007).
4. (Bayardo & Agrawal, 1999) R. Bayardo, R. Agrawal, Mining the most interesting rules, in: Proceedings of the Fifth International Conference on Knowledge Discovery and Data Mining, San Diego, CA, August 1999, pp. 145-154.
5. (Camino et al., 2019) Camino, Ramiro D., Christian A. Hammerschmidt, and Radu State. "Improving Missing Data Imputation with Deep Generative Models." *arXiv preprint arXiv:1902.10666* (2019).
6. (Cazer et al., 2019) C. L. Cazer, M. A. Al-Mamun, K. Kaniyamattam, W. J. Love, J. G. Booth, C. Lanzas, Y. T. Gröhn, Shared multidrug resistance patterns in chicken-associated Escherichia coli identified by association rule mining, Frontiers in Microbiology, 10, 2019, 687.
7. (Cărunta et al., 2019) Alina Cărunta, Mihai Pleşu, Mircea Marin, Antimicrobial resistance patterns detection using gene - EHB 2019, Iasi, Romania, November 21-23, 2019.
8. (Doersch., 2016) Doersch, Carl. "Tutorial on variational autoencoders." *arXiv preprint arXiv:1606.05908* (2016).
9. (Gal & Ghahramani, 2015) Gal, Yarin, and Zoubin Ghahramani. "Bayesian convolutional neural networks with Bernoulli approximate variational inference." *arXiv preprint arXiv:1506.02158* (2015).
10. (Gal & Ghahramani, 2016a) Gal, Yarin, and Zoubin Ghahramani. "Dropout as a bayesian approximation: Representing model uncertainty in deep learning." *international conference on machine learning*. 2016.
11. (Gal & Ghahramani, 2016b) Gal, Yarin, and Zoubin Ghahramani. "A theoretically grounded application of dropout in recurrent neural networks." *Advances in neural information processing systems*. 2016.
12. (Geng & Hamilton, 2006) L. Geng, H. J. Hamilton, Interestingness measures for data mining: a survey, ACM Computing Surveys, 38, no. 3, 2006, article 9.
13. (Hahsler, 2015) M. Hahsler, A Probabilistic Comparison of Commonly Used Interest Measures for Association Rules, 2015, URL: http://michael.hahsler.net/research/association_rules/measures.html
14. (Hahsler et al., 2019) M. Hahsler, C. Buchta, B. Gruen, K. Hornik, arules: Mining association rules and frequent itemsets. R package version 1.6-4, 2019, https://CRAN.R-project.org/package=arules.
15. (Hämäläinen & Nykänen, 2008) W. Hämäläinen, M. Nykänen, Efficient discovery of statistically significant association rules, Eighth IEEE International Conference on Data Mining, 2008, pp. 203-212.

16. (Hämäläinen & Webb, 2019) W. Hämäläinen, G. I. Webb, A tutorial on statistically sound pattern discovery, Data Mining and Knowledge Discovery 33, 2019, pp.325-377.
17. (Helmbold & Philip, 2017) Helmbold, David P., and Philip M. Long. "Surprising properties of dropout in deep networks." *The Journal of Machine Learning Research* 18.1 (2017): 7284-7311.
18. (Kingma & Welling, 2013) Kingma, Diederik P., and Max Welling. "Auto-encoding variational bayes." *arXiv preprint arXiv:1312.6114* (2013).
19. (Lall., 2016) Lall, Ranjit. "How multiple imputation makes a difference." *Political Analysis* 24.4 (2016): 414-433.
20. (Little & Rubin., 2019) Little, Roderick JA, and Donald B. Rubin. *Statistical analysis with missing data*. Vol. 793. John Wiley & Sons, 2019.
21. (Martinez – Ballesteros et al., 2014) M. Martínez-Ballesteros, F. Martínez-Álvarez, A. Troncoso, J. C. Riquelme, Selecting the best measures to discover quantitative association rules, Neurocomputing, 126, 2014, pp. 3-14.
22. (McCoy et al., 2013) McCoy, John T., Steve Kroon, and Lidia Auret. "Variational Autoencoders for Missing Data Imputation with Application to a Simulated Milling Circuit." *IFAC-PapersOnLine* 51.21 (2018): 141-146.
23. (Megiddo & Srikant, 1998) N. Megiddo, R. Srikant, Discovering predictive association rules, in: Proceedings of the Fourth International Conference on Knowledge Discovery and Data Mining, New York, NY: AAAI Press, 1998.
24. (Miok et al., 2019a) Miok, Kristian, Dong Nguyen-Doan, Daniela Zaharie, and Marko Robnik-Šikonja. "Generating Data using Monte Carlo Dropout." *arXiv preprint arXiv:1909.05755* (2019).
25. (Miok et al., 2019b) Kristian Miok, Dong Nguyen-Doan, Marko Robnik-Sikonja and Daniela ZaharieMultiple Imputation for Medical Data using Neural Network-based Autoencoders Combined with Monte Carlo dropout, The 7th IEEE International Conference on E-Health and Bioengineering - EHB 2019, Iasi, Romania, November 21-23, 2019.
26. (Morris et al., 2014) Morris, Tim P., Ian R. White, and Patrick Royston. "Tuning multiple imputation by predictive mean matching and local residual draws." *BMC medical research methodology* 14.1 (2014): 75.
27. (NARMS, 2019) National Antimicrobial Resistance Monitoring System (NARMS) Now: Human Data. Atlanta, Georgia: U.S. Department of Health and Human Services, CDC. 09/18/2019. https://www.cdc.gov/narmsnow Accessed 7/19/2019.
28. (Piatetski – Shapiro, 1991) G. Piatetsky-Shapiro, Discovery, analysis and presentation of strong rules, in: G. Piatetsky-Shapiro, W. Frawley (Eds.), Knowledge Discovery in Databases, MIT Press, Cambridge, MA, 1991, pp. 229-248.
29. (Riondato & Vandin, 2014) M. Riondato, F. Vandin, Finding the true frequent itemsets, Proceedings of the 2014 SIAM International Conference on Data Mining, Philadelphia, USA, April 24-26, 2014, pp. 497-505.
30. (Robnik-Šikonja., 2019) Robnik-Šikonja, Marko. "Dataset comparison workflows." *International Journal of Data Science* 3, no. 2 (2018): 126-145.
31. (Rumelhart et al., 2014) Rumelhart, David E., Geoffrey E. Hinton, and Ronald J. Williams. *Learning internal representations by error propagation*. No. ICS-8506. California Univ San Diego La Jolla Inst for Cognitive Science, 1985.

32. (Tadesse et al., 2012) D.A. Tadesse, S. Zhao, E. Tong, S. Ayers, A. Singh, M.J. Bartholomew et.al., "Antimicrobial drug resistance in Escherichia coli from humans and food animals", United States,1950–2002, Emerging Infectious Diseases, 18 (5), 2012, pp. 741-749.
33. (Tan et al., 2004) P. -N. Tan, V. Kumar, J. Srivastava, Selecting the right objective measure for association analysis, Information Systems, 29, 2004, pp. 293-313.
34. (Tan et al., 2014) P. –N. Tan, M. Steinbach, V. Kumar, Introduction to Data Mining, Pearson Education Ltd, 2014.
35. (Vaillant et al., 2004) B. Vaillant, P. Lenca, S. Lallich, A clustering of interestingness measures, in: Suzuki E., Arikawa S. (eds) Discovery Science. DS 2004. Lecture Notes in Computer Science, vol 3245, Springer, Berlin, Heidelberg, 2004.
36. (Van Buuren., 2018) Van Buuren, Stef. *Flexible imputation of missing data*. Chapman and Hall/CRC, 2018.
37. (Weiß, 2008) C. H. Weiß, Statistical mining of interesting association rules, Statistics and Computing 18, 2008, pp. 185-194.
38. (Zaharia et al., 2019) Claudia Zaharia, Raluca Muresan, Radu Moleriu, Daniela Zaharie, Analysis of association measures used to discover antimicrobial resistance patterns, The 7th IEEE International Conference on E-Health and Bioengineering - EHB 2019, Iasi, Romania, November 21-23, 2019.