

Time series classification

Result outline (Etapa 1)

- SVM obtine cele mai bune rezultate pe cele 3 dataseturi
 - Gradient boosting urmeaza SVM pe dataseturile MITBIH si PTB
 - RandomForest urmeaza SVM pe datasetul RacketSports
- MITBIH + PTB prezinta un nivel de dezechilibru insa rezultatele sunt decente.
- Clasele cel mai bine învățate sunt, de regulă, cele majoritare (destul de bine evidențiat pentru MITBIH sau PTB, unde diferențele absolute în F1 score între clasa cea mai frecventă și cea minoritară pot fi și de 0.15-0.2)
- Puterea predictivă a atributelor
 - RacketSports
 - Tipurile de attribute cele mai predictive care se regasesc in ferestre variate din seria de timp inițială sunt, printre altele: min, max, mean in domeniul timp + signal magnitude area, acceleratie rezultanta, energie, mean in domeniul frecventa
 - Atributele provin din toate axele de măsură într-o proporție similară, cele mai frecvente fiind totuși dim0, dim3 și dim4 (i.e. accelerație pe x, rotație pe x și y)
 - Aceste attribute si axele aferente sunt motivate de natura fiecărui tip de mișcare, cum ar fi intensitatea loviturii, direcțiile de pregătire a loviturii, timing, etc.
 - MITBIH + PTB Diagnostic
 - Tipurile de attribute cele mai predictive care se regasesc in ferestre variate din seria de timp inițială sunt, printre altele: kurtosis, numărul de vârfuri, numărul de valori peste medie, atat în domeniul timp cat și in domeniul frecvență
 - Aceste attribute sunt interconectate prin faptul ca evidențiază abateri de la norma, anomalii, outliers; dată fiind natura sursei de date, are sens selectia acestora deoarece aritmia este o conditie care determina modificări ale formei de unda ECG care accentueaza valorile extreme
- Impactul hiperparametrilor asupra performanței
 - SVM
 - Tipul de kernel are o contribuție mai mare la performanța finală a clasificatorului SVM decât factorul C
 - RandomForest
 - Mai ales din rezultatele obținute pe dataseturile ECG related, reiese că adâncimea maximă influențează cel mai mult performanța modelului
 - Numărul de estimatori urmează adâncimea ca relevanță însă este necesar ca și max_samples să fie suficient de mare pentru ca fiecare estimator să observe un subset reprezentativ pentru intregul dataset si să evite o dispersie/diversitate prea mare.

- Gradient boosted trees
 - Learning rate influențează semnificativ performanța, mai ales pentru datele de dimensionalitate mare(i.e. RacketSports)
 - Numărul de estimatori urmează learning rate ca relevanță însă și max_depth trebuie să fie suficient de mare pentru a putea surprinde patterns in date, pentru a nu se afla in regim de underfitting și pentru a putea generaliza.

Result outline (Etapa 2)

- Pentru atingerea unei performante cat mai buna modelelor de invatare profunda, s-a constatat ca este necesar un numar mai mare de epoci(pana la aproximativ 15-20, dupa care se intra intr-o zona de diminishing returns sau de overfit dupa diferite cazuri arhitecturale, cum se poate vedea in figurile de train/test loss)
- Exista o valoare optimă a batch size prin care se balanseaza eficienta antrenarii si performanta finala
 - Marind aceasta valoare, performanta incepe sa se degradeze, cu diferente notabile in ceea ce priveste clasele minoritare(e.g. clasele de index 1 si 3 in MITBIH sau clasa 0/normal in PTB)
- MITBIH
 - MLP - Top accuracy: 97.8 , top F1 score minority class 3: 74.6
 - Conv1D - Top accuracy: **98.1** , top F1 score minority class 3: 73.6
 - LSTM - Top accuracy: 97.7 , top F1 score minority class 3: **76**
- PTB
 - MLP - Top accuracy: 96.5 , top F1 score minority class 0: 93.6
 - **Conv1D** - Top accuracy: **97.3** , top F1 score minority class 0: **95**
 - LSTM - Top accuracy: 96.3 , top F1 score minority class 0: 93.3
- Niciunul din rezultatele de top nu a fost obtinut folosind un learning rate scheduler
 - Grad de influenta observat ca fiind scazut
 - In experimente, optiuni precum CosineAnnealingLR sau LinearLR au produs rezultate comparabile inasa, prin comparatie, inferioare
 - Posibile cauze: linear ramp-up prea scurt pentru a fi relevant(LinearLR), perioade prea lungi de learning rate relativ mic(CosineAnnealingLR)
- Conv1D ating acuratetea maxima
 - Justificari:
 - utilizeaza eficient pattern-uri locale din seriile de timp
 - cu suficiente straturi de feature extraction(e.g. configuratia optima are [32,64,128,256]), se pot desprinde caracteristici ierarhice/globale care pot fi distinse mai usor in partea de clasificare
- MLP pot atinge rezultate respectabile cu doar 2 straturi ascunse
 - Fiecare sample de intrare nu are o dimensionalitate foarte mare(187 timesteps/features) asa ca se pot utiliza straturi putine (e.g. [128,32])
- LSTM au atins rezultatele maxime in 2 configuratii diferite pentru fiecare dataset
 - pentru MITBIH, a fost suficient sa se foloseasca output state aferent ultimului time step si un batch size de 256 (#1)

- pentru PTB, feature vector transmis clasificatorului MLP a fost format prin mean pooling asupra tuturor output states + batch size 32 (#2)
- cel mai probabil, varianta de LSTM pentru PTB functioneaza decent si pe MITBIH
- motivul pentru care cele 2 abordari sunt valide pentru fiecare caz in parte este reprezentat de diferentele de dimensiune ale celor 2 surse de date
 - varianta #1 filtreaza(discards) mai multa informatie din fiecare exemplu dar retine suficient din aceasta + are destule exemple pentru a invata
 - varianta #2 are la dispozitie mai putine exemple, drept urmare este necesar mai mult context pentru fiecare exemplu
 - cardinalitatea datasetului este acelasi motiv pentru care o valoare de dropout nenula apare in varianta #2
- Rezultatele de top s-au putut atinge si fara varianta bidirectionala(e.g. best configuration LSTM MITBIH), dar cu num_layers=2
 - Posibila justificare: caracteristicile unidirectionale sunt suficiente pentru a distinge diferitele niveluri de aritmie; pentru clasificarea binara normal/anormal in schimb, este necesara mai multa informatie
 - + diferenta de cardinalitate

1. RacketSports

Descriere

- RacketSports este un set de date ce conține înregistrări de smart watch făcute în timpul unui joc de badminton sau squash.
- Datele descriu semnale de accelerație (pe axele x, y și z) și giroscop (rotație pe axele x, y și z), surprinzând înregistrări de 3 secunde etichetate ca reprezentând tipul de mișcare din joc (forehand / rever pentru squash și clear / smash pentru badminton).

Exploratory Data Analysis

- Dimensiune training set: 151 secvențe multivariate(6 dimensiuni temporale)
- Dimensiune test set: 152 secvențe

Dataset balance

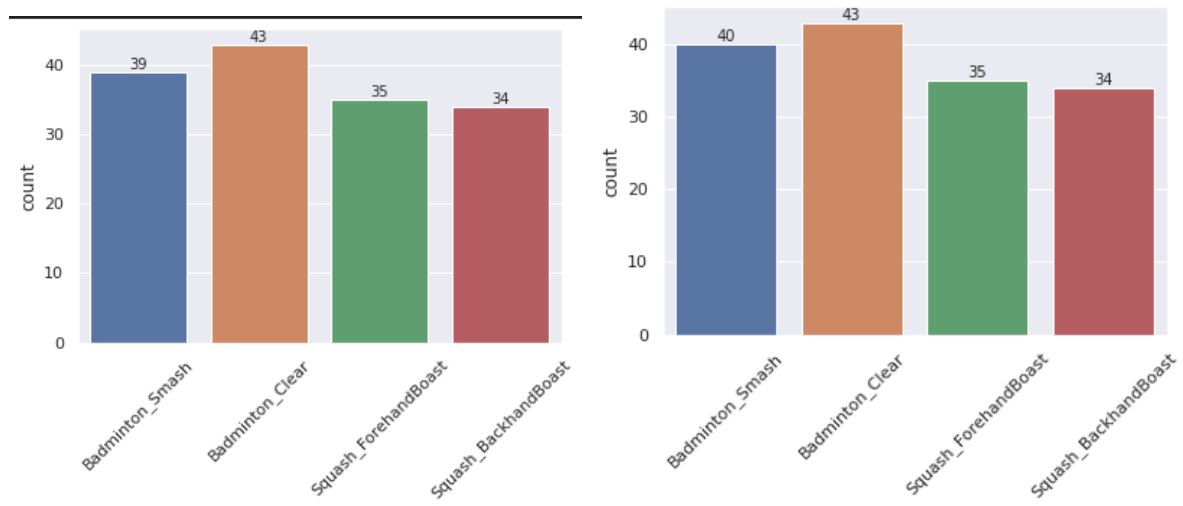
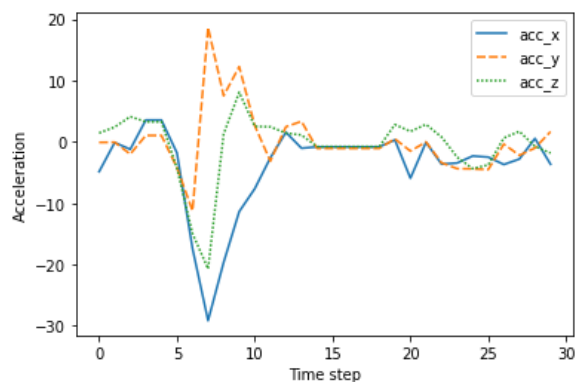


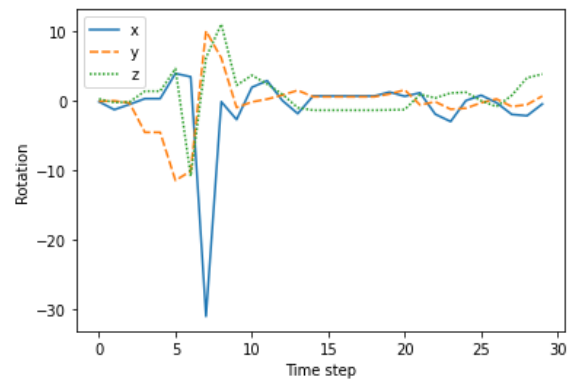
Figura 1. Frecvența de apariție a fiecărei clase în setul de date
(Left) Antrenare (Right) Testare

- In Figura 1, se observa ca setul de date este echilibrat in ambele split-uri (i.e. imbalance factor identic si apropiat de 1.0) si, de asemenea, nu exista distribution shift la nivelul claselor.
 - Măsurile de tratare a class imbalance nu sunt necesare in acest context.

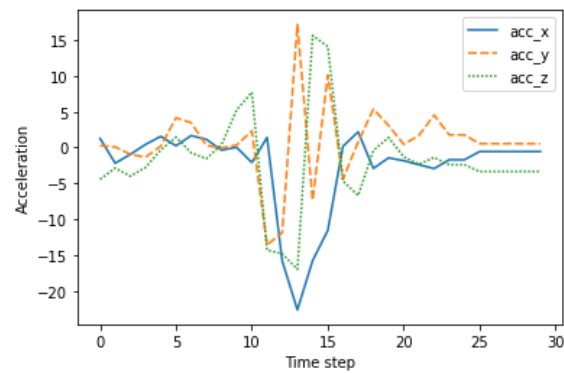
Exemple corespunzătoare tipurilor de mișcări



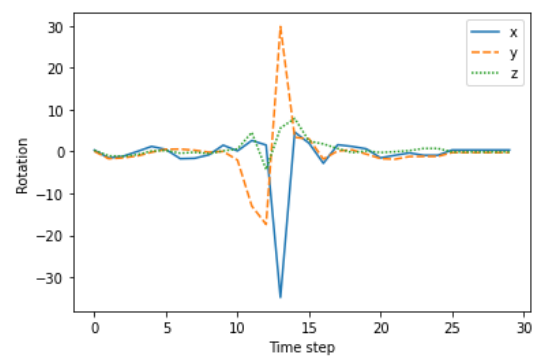
(a)



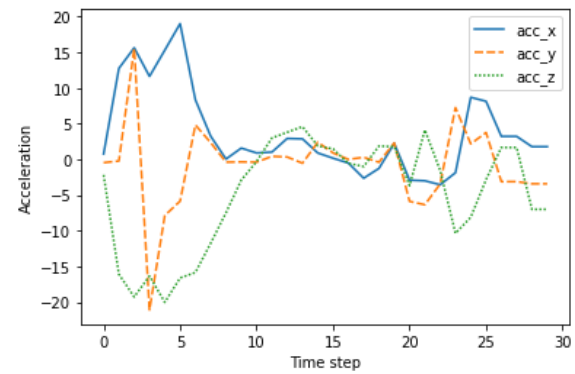
(b)



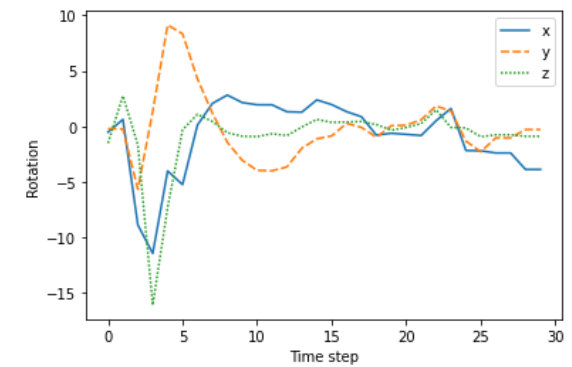
(c)



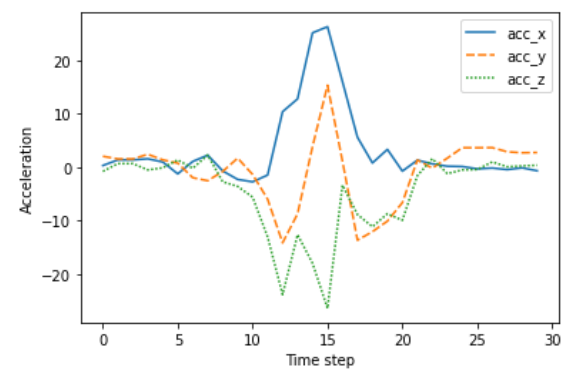
(d)



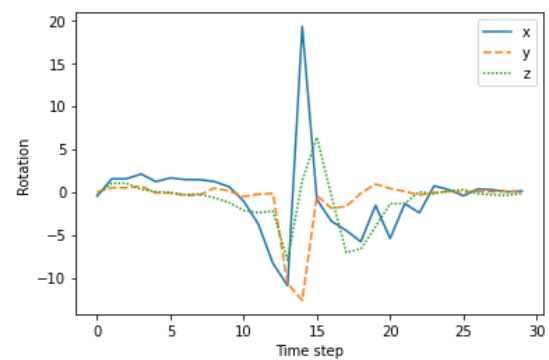
(e)



(f)



(g)



(h)

Figura 2. (a-b) Badminton Clear (c-d) Badminton Smash (e-f) Squash Backhand (g-h) Squash Forehand

Distribution plots per axă și acțiune

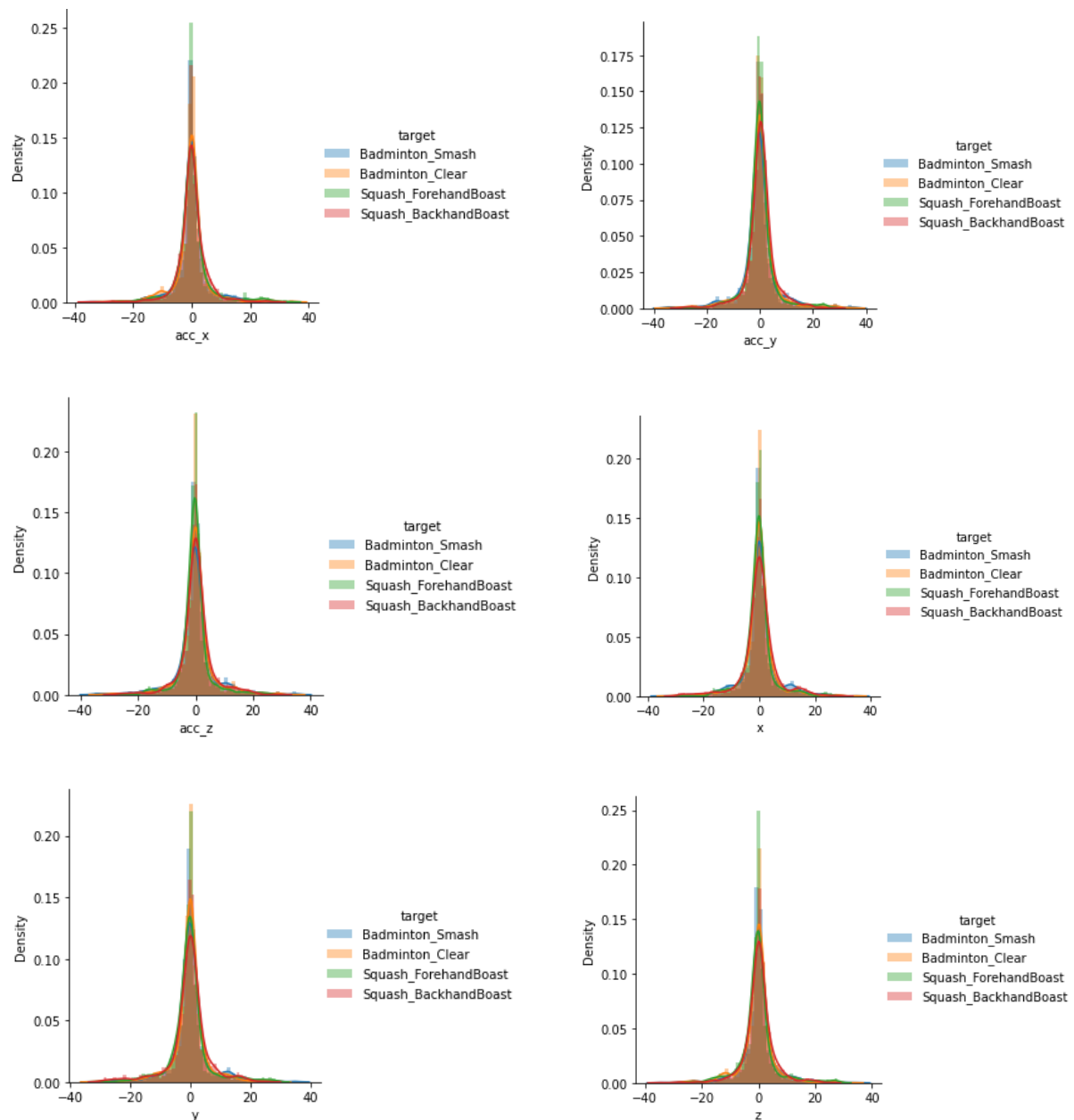


Figura 3. Distribuția valorilor înregistrate de senzori per fiecare axă, evidențiate separat în funcție de tipul de acțiune

- În Figura 3, se evidențiază faptul că nu există un șablon clar de diferențiere între distribuțiile de valori per axă, în funcție de gestul executat. De aceea, mai multe caracteristici trebuie construite pentru a putea clasifica tipurile de mișcări.

Classical ML

Feature extraction & selection

- Caracteristicile extrase provin din attributele statistice aplicate asupra unor ferestre cu overlap pe fiecare din cele 6 dimensiuni (window_size=10, window_step=3)
- Total features: 1372 -> VarianceThreshold(2.0) -> 902 utilizate la antrenare

SVM

Best configuration

C	kernel
5.0	rbf

Rezultate cross validation

Rezultatele complete se regasesc in ***racketsports_svm_cv.xlsx***

Best config results:

Metric name	Value(mean, std across folds)
Accuracy	0.669, 0.1
Precision (macro)	0.725, 0.085
Recall (macro)	0.677, 0.103
F1 (macro)	0.682, 0.099

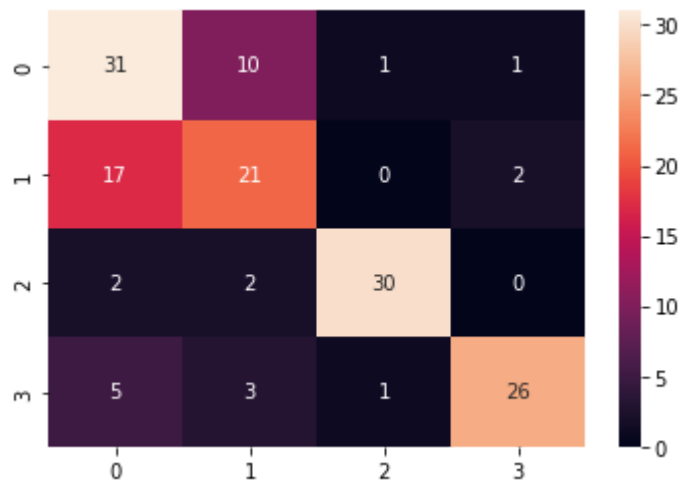
Rezultate test set

Rezultatele complete se regasesc in ***racketsports_svm_results.xlsx***

Best config results:

Metric name	Value(mean, std across classes)
Accuracy	0.711
Precision	0.745, 0.173
Recall	0.718, 0.127
F1	0.727, 0.141

Confusion matrix



RandomForest

Best configuration

n_estimators	max_depth	max_samples
220	8	1.0

Rezultate cross validation

Rezultatele complete se regasesc in ***racketsports_rforest_cv.xlsx***

Best config results:

Metric name	Value(mean, std across folds)
Accuracy	0.695, 0.028
Precision (macro)	0.744, 0.039
Recall (macro)	0.702, 0.038
F1 (macro)	0.705, 0.032

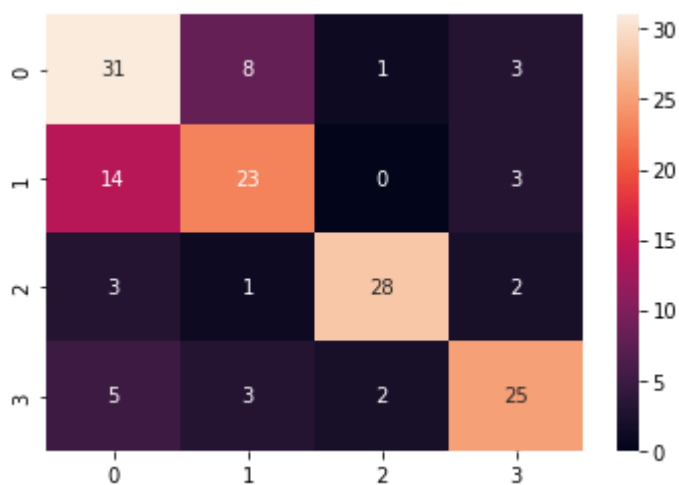
Rezultate test set

Rezultatele complete se regasesc in ***racketsports_rforest_results.xlsx***

Best config results:

Metric name	Value(mean, std across classes)
Accuracy	0.658
Precision	0.678, 0.113
Recall	0.658, 0.1
F1	0.663, 0.091

Confusion matrix



Gradient boosted trees

Best configuration

n_estimators	max_depth	learning_rate
40	5	0.1

Rezultate cross validation

Rezultatele complete se regasesc in ***racketsports_xgb_cv.xlsx***

Best config results:

Metric name	Value(mean, std across folds)
Accuracy	0.648, 0.077
Precision (macro)	0.685, 0.058
Recall (macro)	0.659, 0.086
F1 (macro)	0.654, 0.077

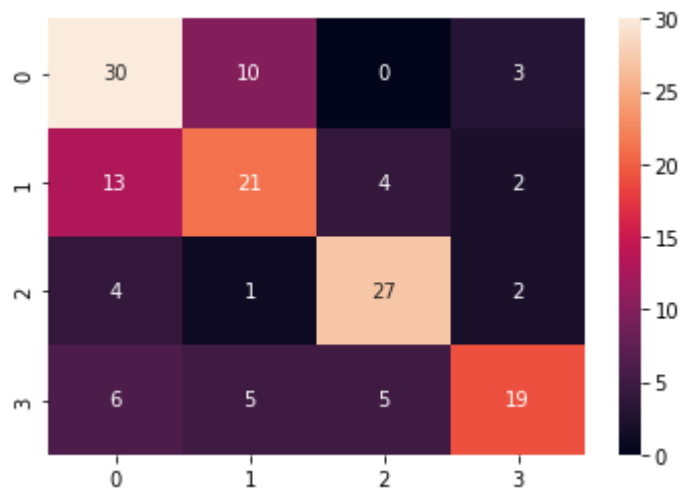
Rezultate test set

Rezultatele complete se regasesc in ***racketsports_xgb_results.xlsx***

Best config results:

Metric name	Value(mean, std across classes)
Accuracy	0.638
Precision	0.654, 0.087
Recall	0.64, 0.112
F1	0.641, 0.082

Confusion matrix



MIT-BIH

Descriere

- Setul de date MIT-BIH Arrhythmia propune un task de clasificare in 5 clase (tipuri de aritmii)
- Semnalele corespund formelor electrocardiografei (ECG) ale băților inimii pentru cazul normal și cazurile afectate de diferite aritmii și infarct miocardic. Aceste semnale sunt preprocesate și segmentate, fiecare segment corespunzând unei băți de inimă.

Exploratory Data Analysis

- Dimensiune training set: 87554 secvențe univariate(187 de segmente, cu posibil padding)
- Dimensiune test set: 21892 secvențe

Dataset balance

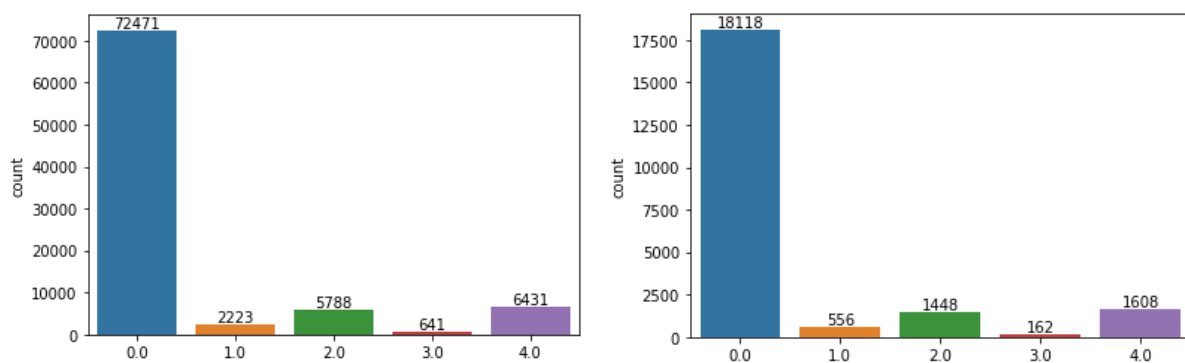


Figura 4. Frecvența de apariție a fiecărei clase în setul de date
(Left) Antrenare (Right) Testare

- In Figura 4, se observa ca setul de date este foarte dezechilibrat in ambele split-uri (i.e. imbalance factor identic si aproximativ 111) și, de asemenea, nu exista distribution shift la nivelul claselor.
 - Măsurile de tratare a class imbalance pot fi utile in acest caz in vederea imbunatatirii performantei clasei/claselor minoritare.

Exemple corespunzătoare tipurilor de aritmie

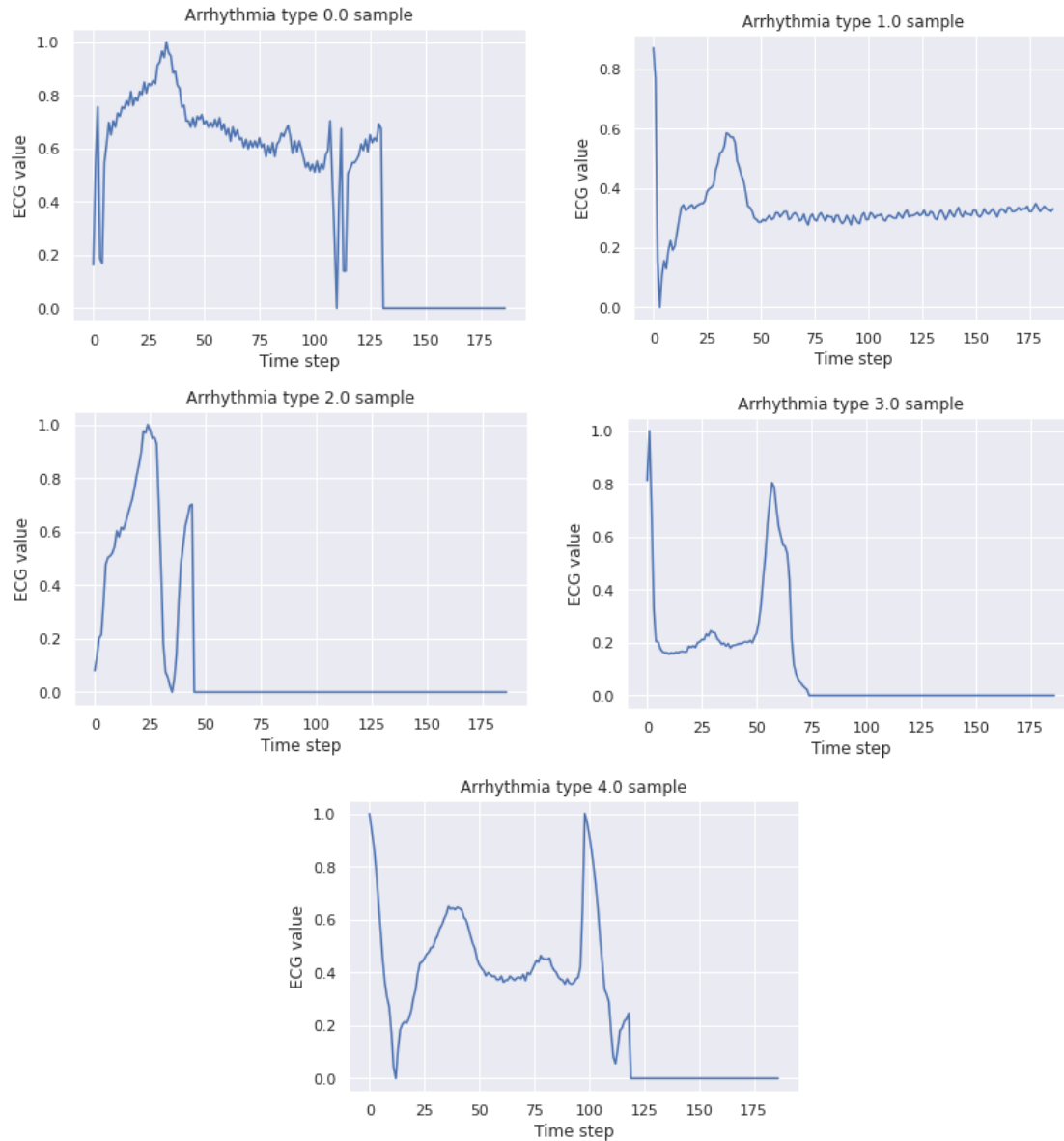
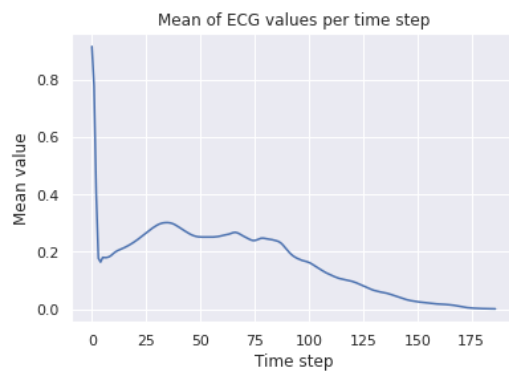
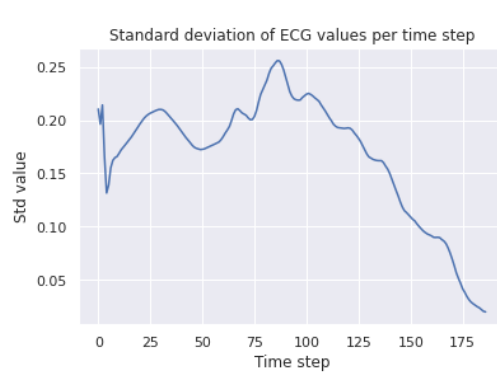


Figura 5. Exemple pentru fiecare tip de aritmie. Type 0 corespunde unui semnal normal, doar restul claselor reprezintă condiții medicale anormale. Padding-ul adăugat aduce în același interval de timp semnale de durate diverse

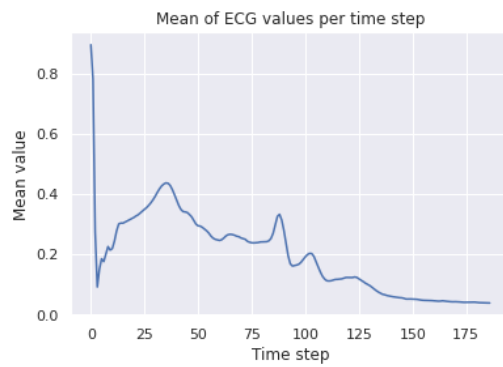
Statistici globale per time step (medie și deviație standard)



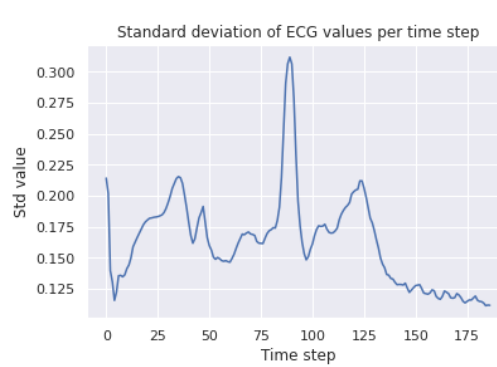
(a)



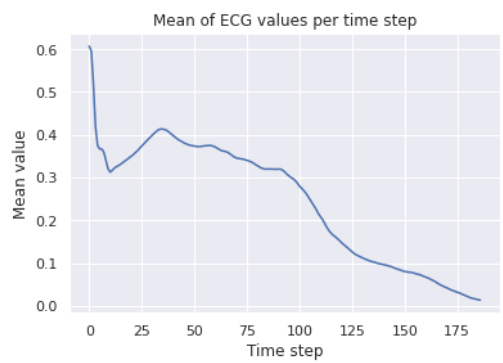
(b)



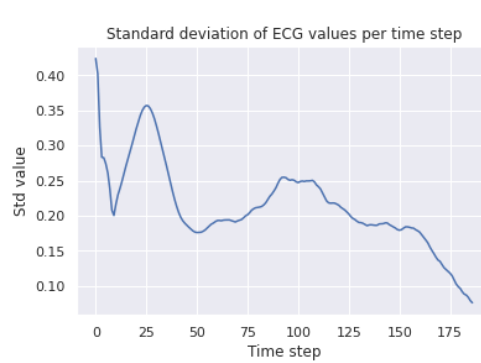
(c)



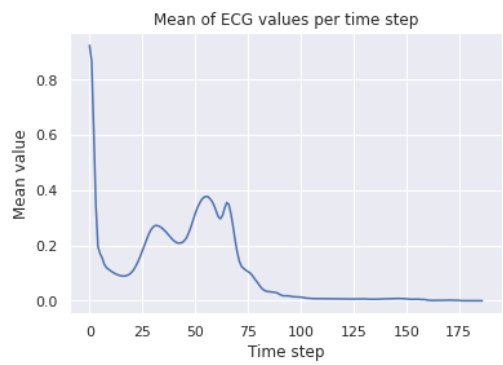
(d)



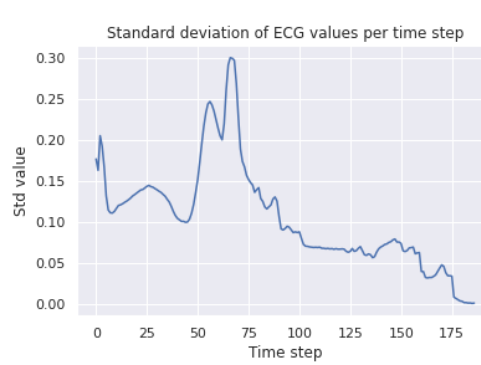
(e)



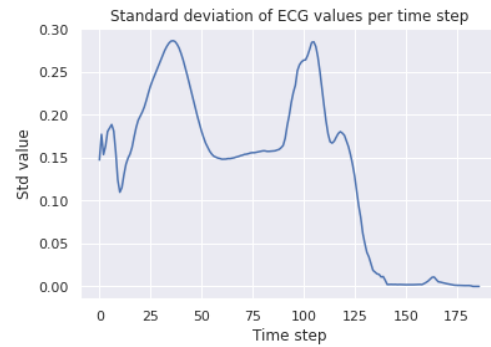
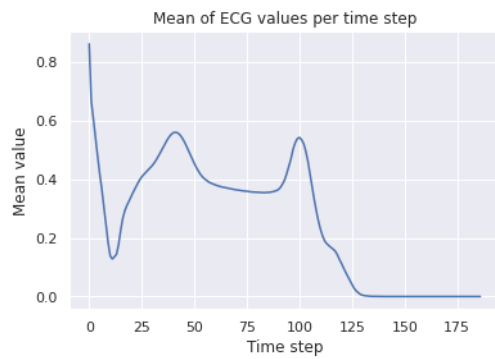
(f)



(g)



(h)



(i)

(j)

Figura 6. Media și deviația standard per time step pentru fiecare tip de semnal:

(a-b) Type 0 (c-d) Type 1 (e-f) Type 2 (g-h) Type 3 (i-j) Type 4

- In Figura 6, se constată diferențe notabile între semnalul mediu și deviațiile standard aferente fiecărei clase. Aceste 2 caracteristici ,combinat cu un set mai extins de atribute cu scopul de a putea descrie și diferenția mai bine instanțele, pot reprezenta predictorii buni ai tipului de aritmie.

Classical ML

Feature extraction & selection

- Caracteristicile extrase provin din atributele statistice aplicate asupra unor ferestre cu overlap pe seria de valori ECG(window_size=20, window_step=8)
- Total features: 756 -> VarianceThreshold(2.0) -> 74 utilizate la antrenare

SVM

Best configuration

C	kernel
10.0	rbf

Rezultate cross validation

Rezultatele complete se regasesc in ***mitbih_svm_cv.xlsx***

Best config results:

Metric name	Value(mean, std across folds)
Accuracy	0.977, 0.001
Precision (macro)	0.946, 0.005
Recall (macro)	0.847, 0.008
F1 (macro)	0.89, 0.007

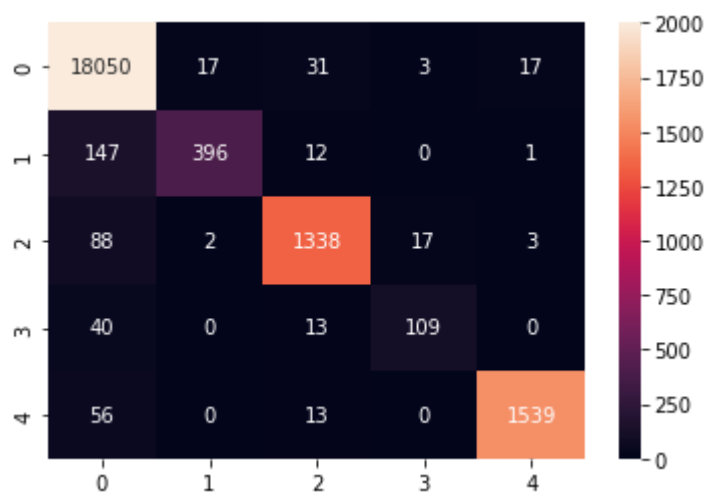
Rezultate test set

Rezultatele complete se regasesc in ***mitbih_svm_results.xlsx***

Best config results:

Metric name	Value(mean, std across classes)
Accuracy	0.979
Precision	0.944, 0.051
Recall	0.852, 0.133
F1	0.893, 0.094

Confusion matrix



RandomForest

Best configuration

n_estimators	max_depth	max_samples
220	8	1.0

Rezultate cross validation

Rezultatele complete se regasesc in ***mitbih_rforest_cv.xlsx***

Best config results:

Metric name	Value(mean, std across folds)
Accuracy	0.955, 0.001
Precision (macro)	0.955, 0.006
Recall (macro)	0.693, 0.006
F1 (macro)	0.780, 0.006

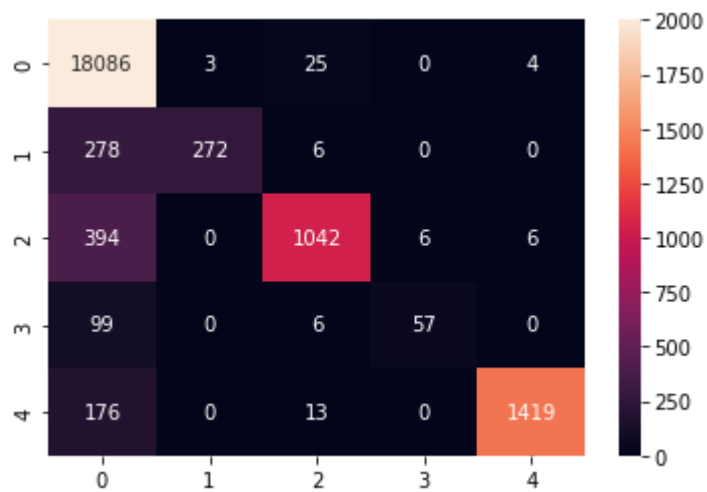
Rezultate test set

Rezultatele complete se regasesc in ***mitbih_rforest_results.xlsx***

Best config results:

Metric name	Value(mean, std across classes)
Accuracy	0.942
Precision	0.776, 0.389
Recall	0.572, 0.363
F1	0.636, 0.356

Confusion matrix



Gradient boosted trees

Best configuration

n_estimators	max_depth	learning_rate
100	8	0.1

Rezultate cross validation

Rezultatele complete se regasesc in ***mitbih_xgb_cv.xlsx***

Best config results:

Metric name	Value(mean, std across folds)
Accuracy	0.974, 0.001
Precision (macro)	0.953, 0.007
Recall (macro)	0.820, 0.008
F1 (macro)	0.875, 0.007

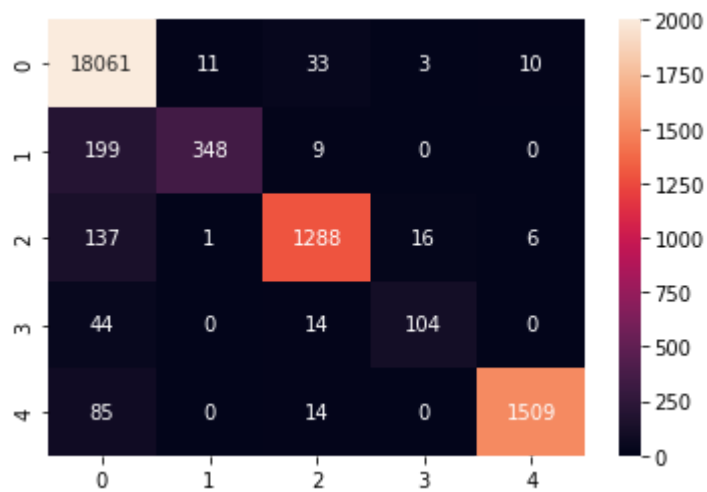
Rezultate test set

Rezultatele complete se regasesc in ***mitbih_xgb_results.xlsx***

Best config results:

Metric name	Value(mean, std across classes)
Accuracy	0.973
Precision	0.945, 0.051
Recall	0.819, 0.155
F1	0.871, 0.106

Confusion matrix



Deep Learning

MLP

Best configuration (architecture + optimization setup)

Parameter	Value
hidden_layers	2
n_units(hidden sizes from input to output)	[128, 32]
dropout	0.0
optimizer	Adam
learning rate	1e-3
batch size	256
epochs	20
scheduler	None
weight decay	None

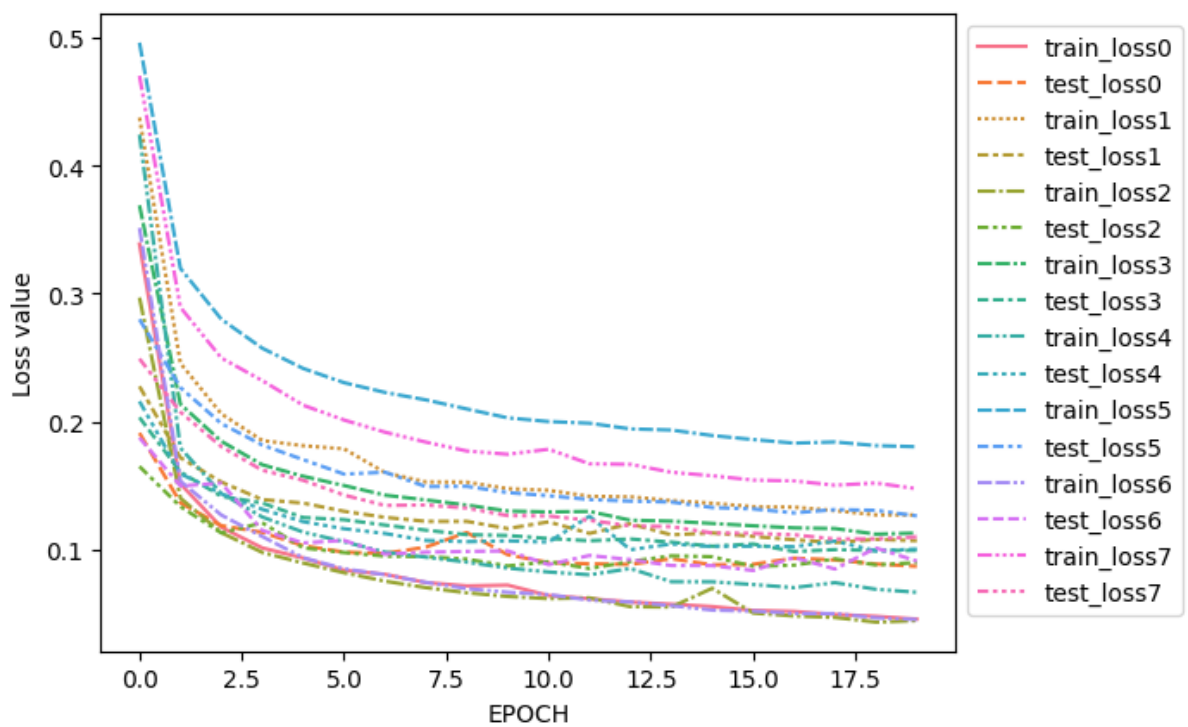
Rezultate best configuration

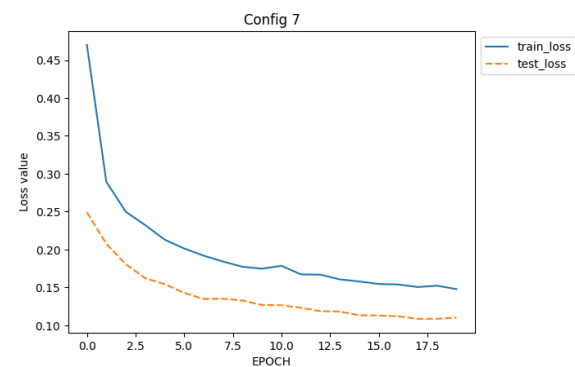
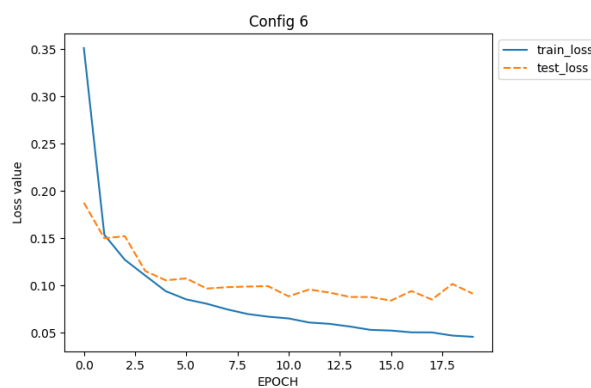
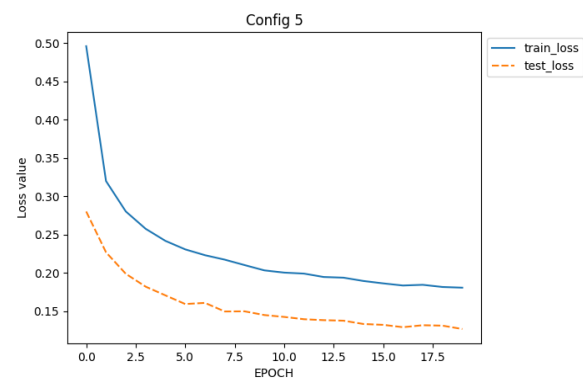
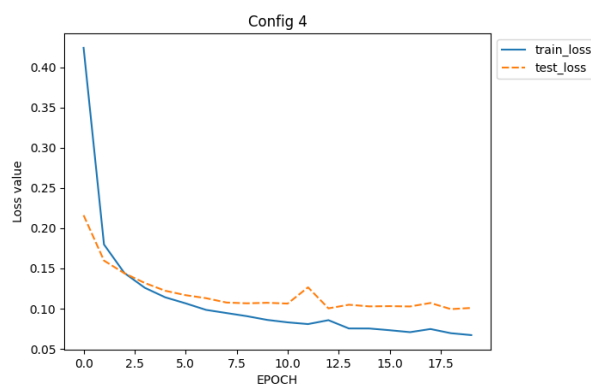
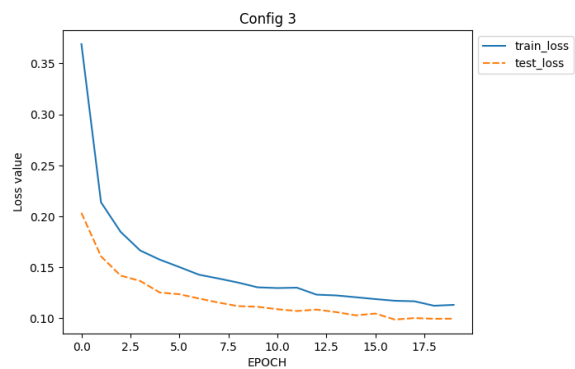
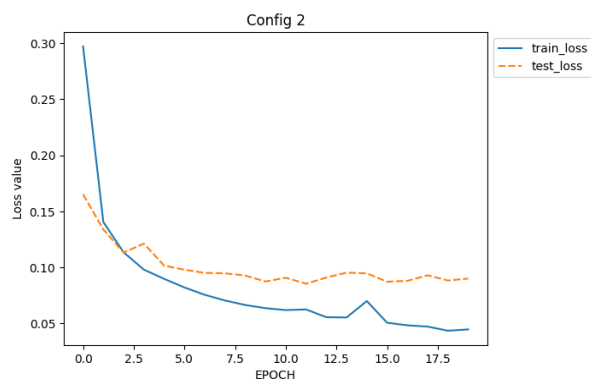
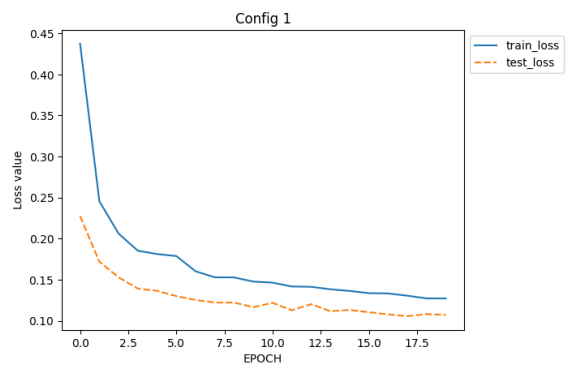
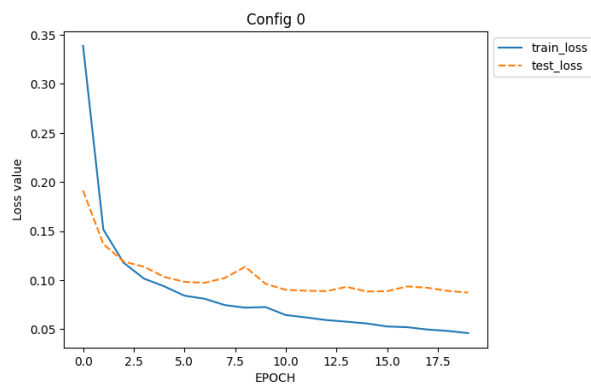
Metric name	Value
Accuracy	97.8
0_precision	98.2
0_recall	99.6
0_f1	98.9
1_precision	89.3
1_recall	69.1
1_f1	77.9
2_precision	95.8
2_recall	91.4

2_f1	93.5
3_precision	82.7
3_recall	67.9
3_f1	74.6
4_precision	98.5
4_recall	96.8
4_f1	97.6

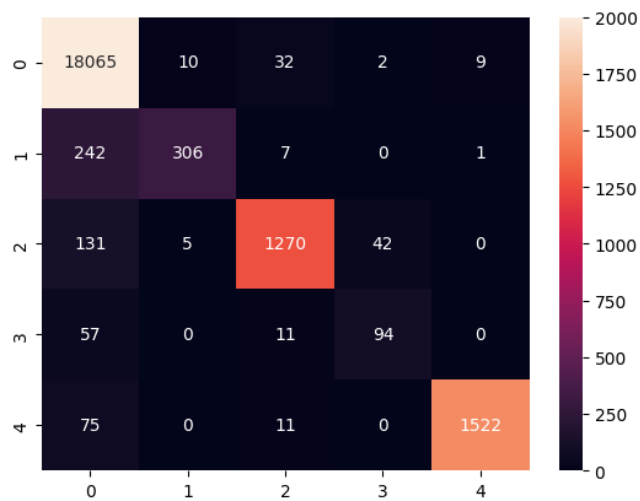
Loss evolution

Cele 8 configuratii corespund celor evidentiata din mlp_results.xlsx, fila MITBIH, in ordinea in care apar





Confusion matrix



Convolutional Neural Network

Best configuration (architecture + optimization setup)

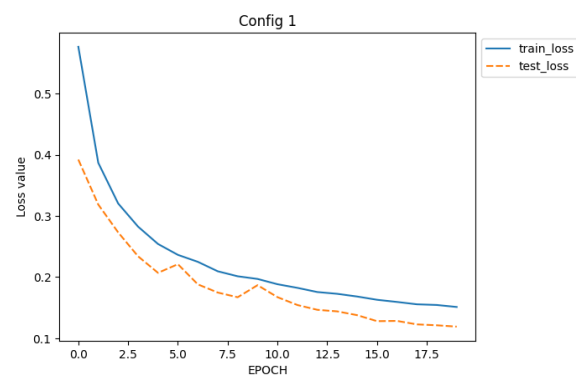
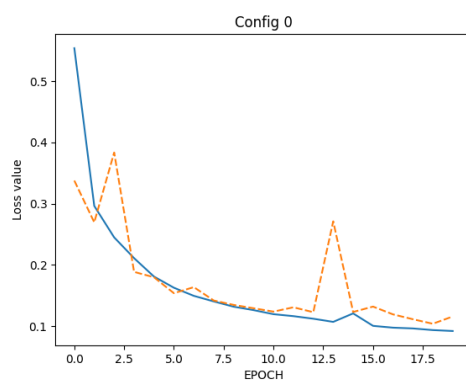
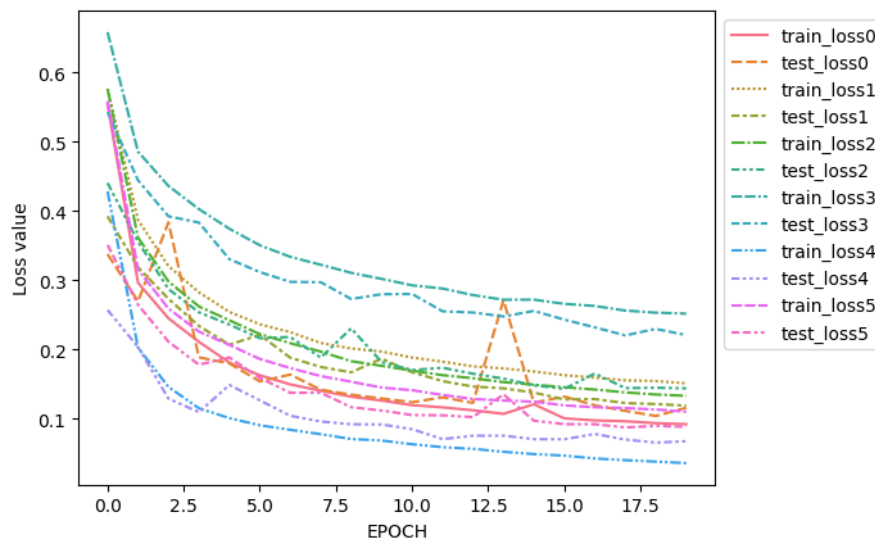
Parameter	Value
hidden_layers(CNN feature extractor part)	4
cnn_sizes	[32,64,128,256]
MLP classifier	[64,16], no dropout
dropout	0.0
optimizer	Adam
learning rate	1e-3
batch size	128
epochs	20
scheduler	None
weight decay	None

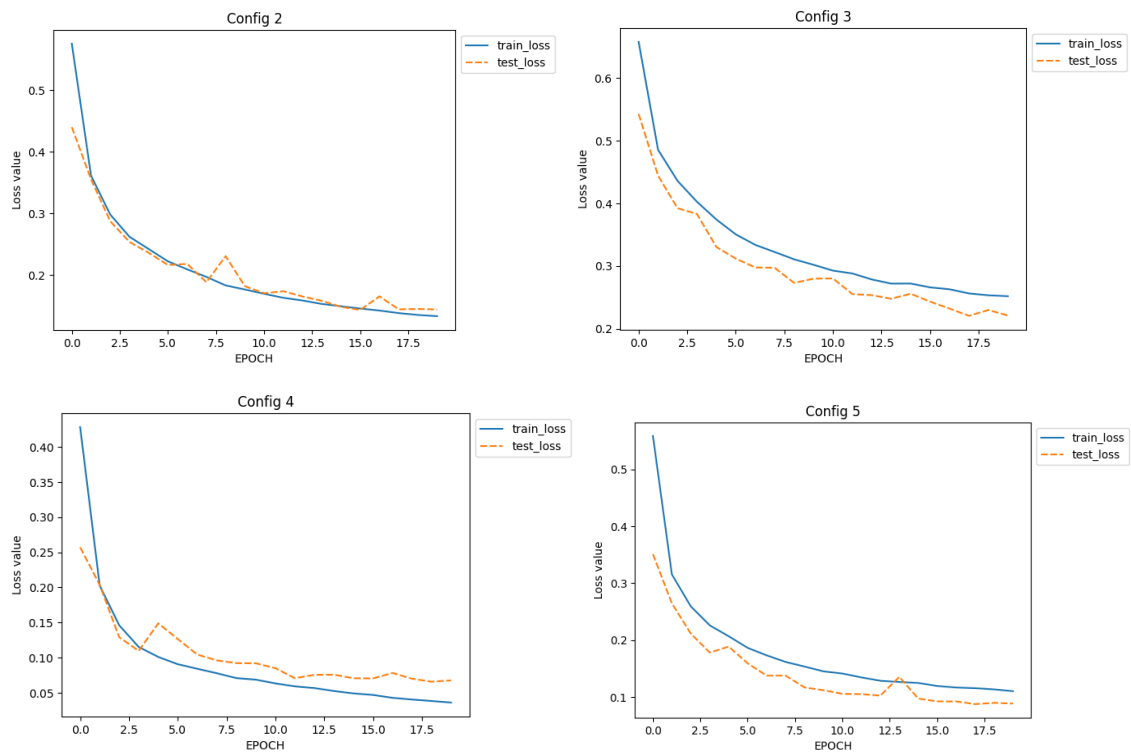
Rezultate best configuration

Metric name	Value
Accuracy	98.1
0_precision	98.8
0_recall	99.3
0_f1	99
1_precision	95.3
1_recall	65.6
1_f1	77.7
2_precision	93
2_recall	96.5
2_f1	94.7
3_precision	73.2
3_recall	74.1
3_f1	73.6
4_precision	98.6
4_recall	99.3
4_f1	98.9

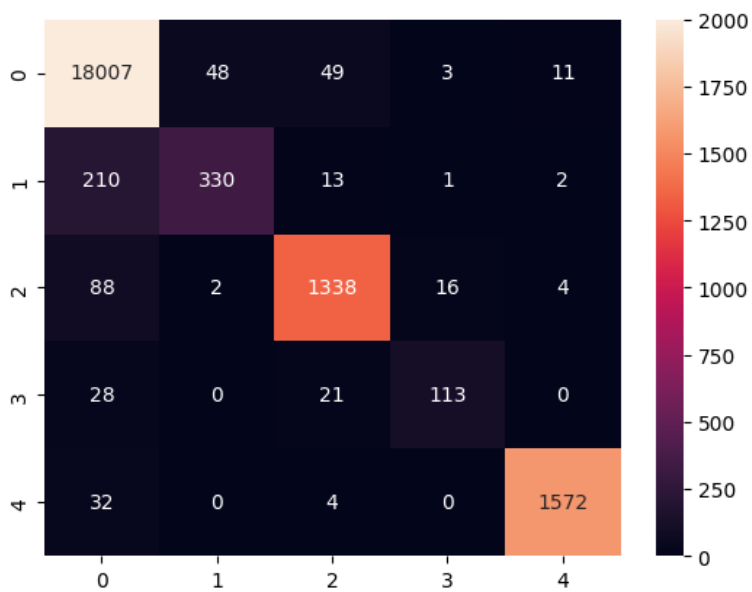
Loss evolution

Cele 6 configuratii corespund celor evidentiata din conv_results.xlsx, fila MITBIH, in ordinea in care apar





Confusion matrix



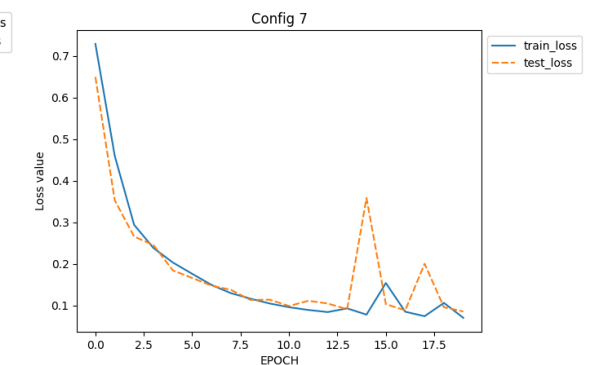
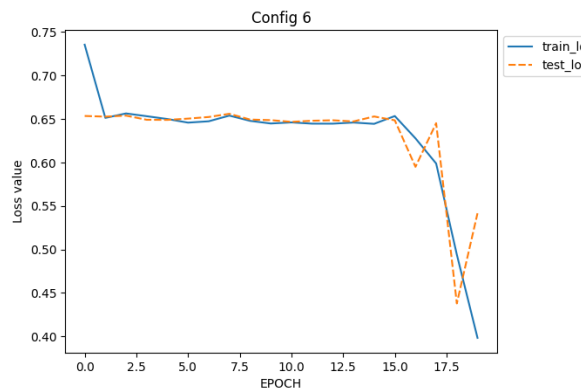
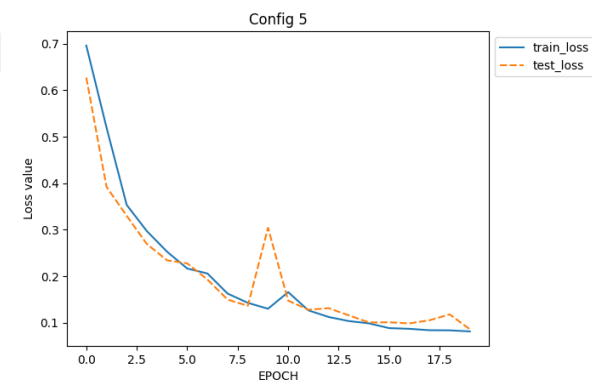
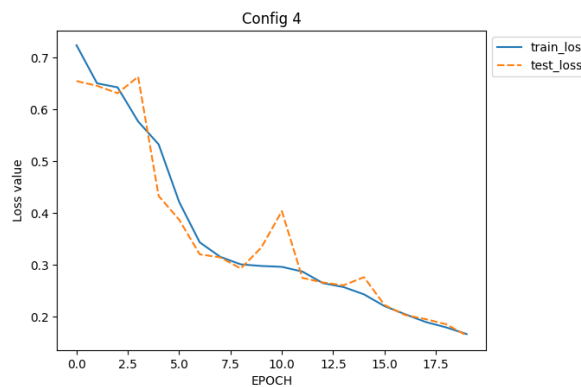
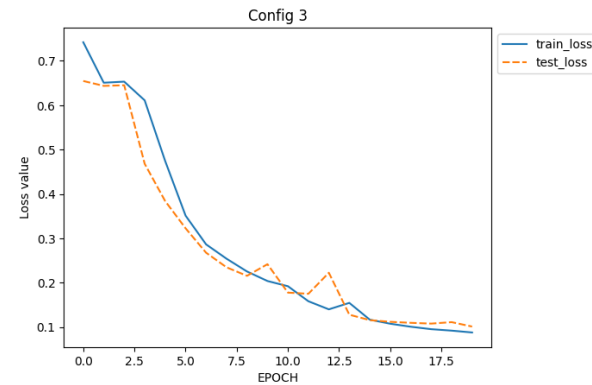
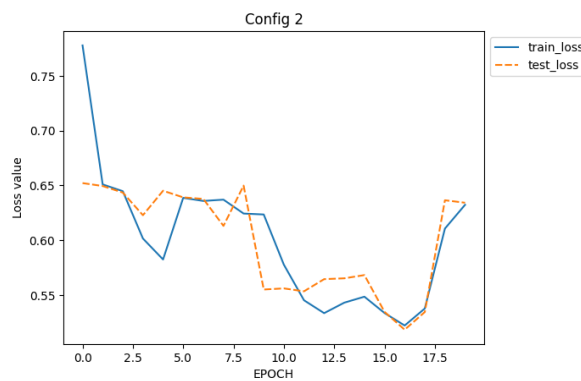
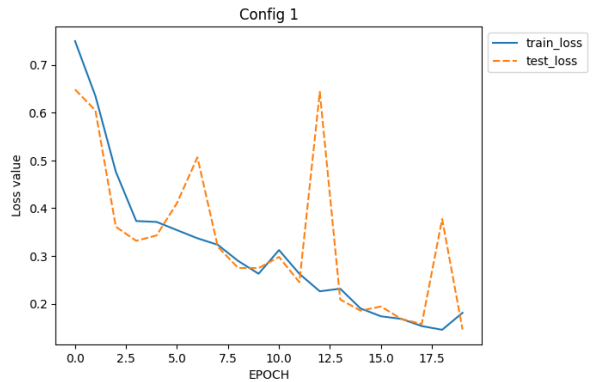
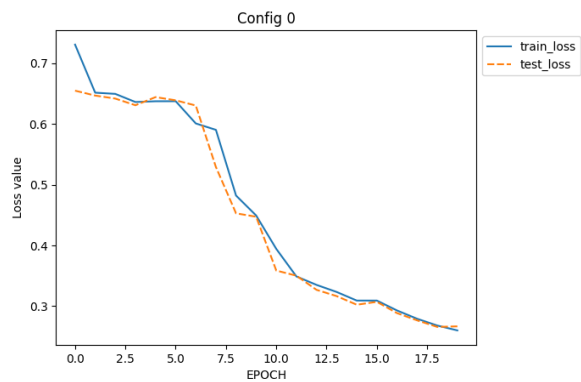
Recurrent Neural Network

Best configuration (architecture + optimization setup)

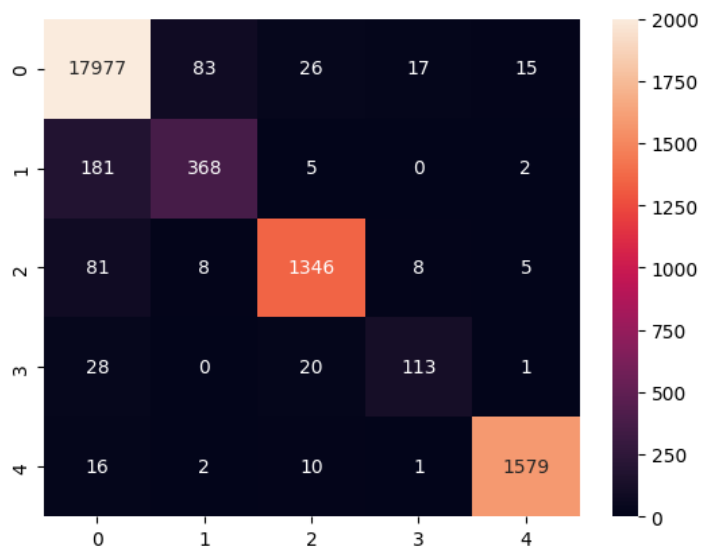
Parameter	Value
hidden_size	128
bidirectional	False
num_layers	2
dropout	0.0
MLP classifier	[32], no dropout
optimizer	Adam
learning rate	1e-3
batch size	256
epochs	20
scheduler	None
weight decay	None
other	Only last output state passed further to MLP

Rezultate best configuration

Metric name	Value
Accuracy	97.7
0_precision	98.2
0_recall	99.4
0_f1	98.8
1_precision	89.1
1_recall	62.9
1_f1	73.8
2_precision	92.8
2_recall	94.5



Confusion matrix



PTB Diagnostic

Descriere

- Setul de date PTB Diagnostic propune o clasificare binara (bătăie normala sau anormala a inimii)
- Semnalele corespund formelor electrocardiografei (ECG) ale bătăilor inimii pentru cazul normal și cazurile afectate de diferite aritmii și infarct miocardic. Aceste semnale sunt preprocesate și segmentate, fiecare segment corespunzând unei bătăi de inimă.

Exploratory Data Analysis

- Numar exemple
 - normale: 4046
 - anormale: 10506
- Dimensiune training set: 80% din numărul total de exemple(secvențe univariate de 187 de segmente, cu posibil padding)
- Dimensiune test set: 20% din numărul total de exemple

Dataset balance

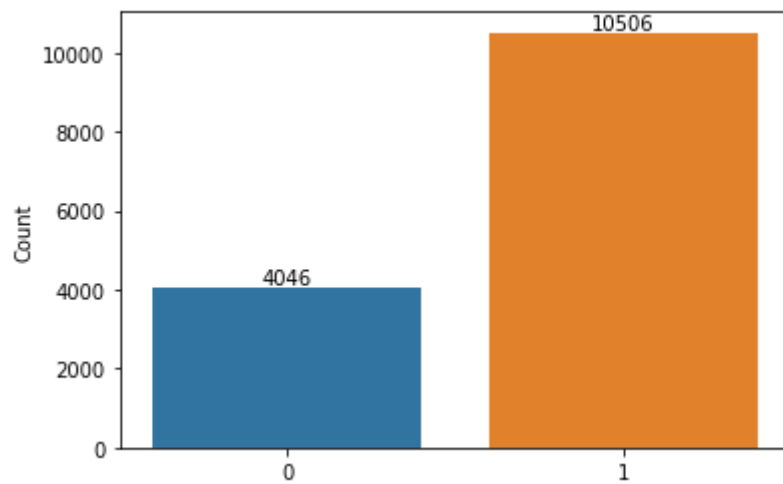


Figura 7. Frecvența de apariție a fiecărei clase în setul de date

- In Figura 7, se observa ca setul de date este destul de dezechilibrat (i.e. imbalance factor aproximativ 2.6)
 - Măsurile de tratare a class imbalance pot fi utile in acest caz in vederea imbunatatirii performantei clasei minoritare insa dezechilibrul nu este atat de pronuntat incat sa nu permita o performanta destul de buna si fara aplicarea de tehnici suplimentare.

Exemple corespunzătoare tipurilor de aritmie

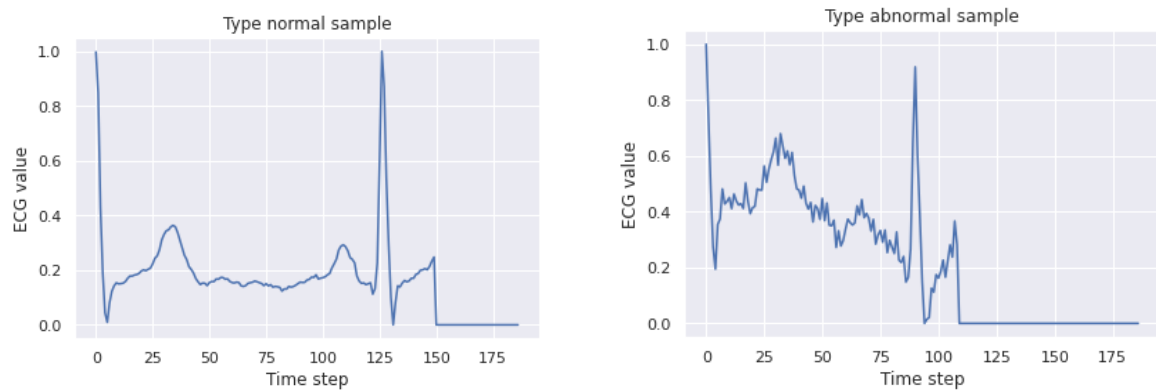
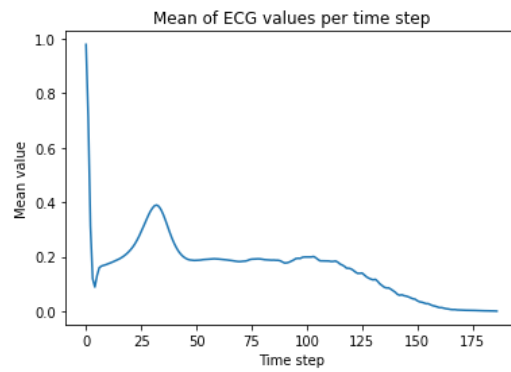
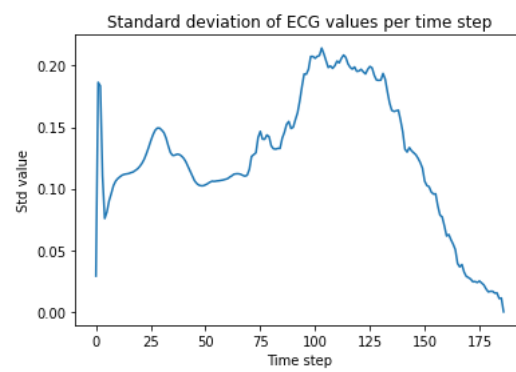


Figura 8. Exemple pentru fiecare tip de semnal ECG.

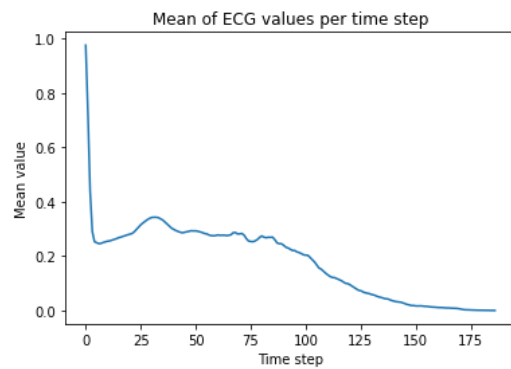
Statistici globale per time step (medie și deviație standard)



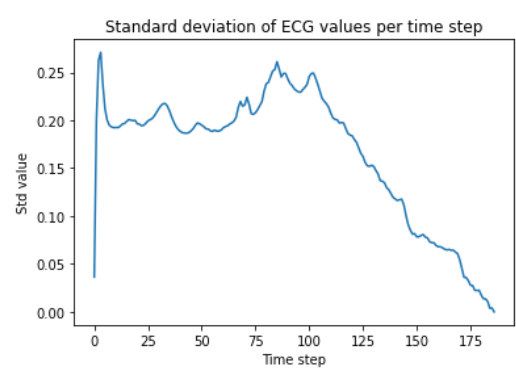
(a)



(b)



(c)



(d)

Figura 9. Media și deviația standard per time step pentru fiecare tip de semnal:
(a-b) normal (c-d) anormal

- In Figura 9, se observa ca tipurile normal si anormal au proprietati apropiate insa cu anumite diferente (e.g. inaltimea varfului centrat aproape de pasul 25, lungimea platoului din deviatia standard, panta de descrestere a deviatiei standard, etc.) care ar putea fi detectate si utilizate de un clasificator.

Classical ML

SVM

Best configuration

C	kernel
10.0	rbf

Rezultate cross validation

Rezultatele complete se regasesc in ***ptbdb_svm_cv.xlsx***

Best config results:

Metric name	Value(mean, std across folds)
Accuracy	0.969, 0.001
Precision (macro)	0.963, 0.003
Recall (macro)	0.961, 0.002
F1 (macro)	0.962, 0.001

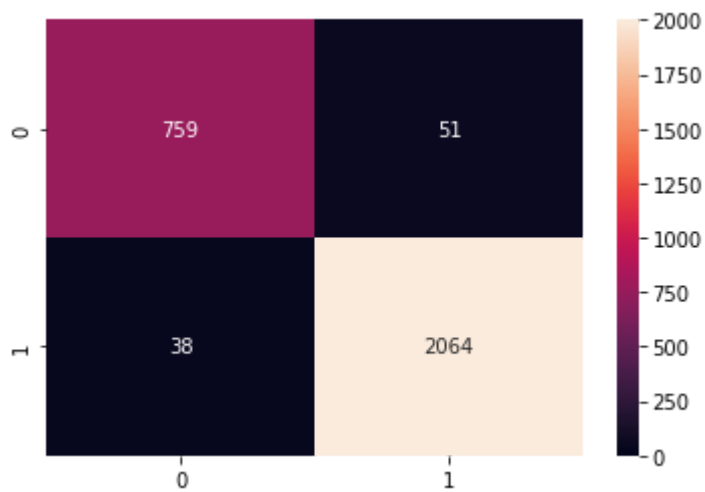
Rezultate test set

Rezultatele complete se regasesc in ***ptbdb_svm_results.xlsx***

Best config results:

Metric name	Value(mean, std across classes)
Accuracy	0.969
Precision	0.964, 0.012
Recall	0.959, 0.022
F1	0.962, 0.017

Confusion matrix



RandomForest

Best configuration

n_estimators	max_depth	max_samples
160	12	1.0

Rezultate cross validation

Rezultatele complete se regasesc in ***ptbdb_rforest_cv.xlsx***

Best config results:

Metric name	Value(mean, std across folds)
Accuracy	0.958, 0.004
Precision (macro)	0.959, 0.005
Recall (macro)	0.936, 0.006
F1 (macro)	0.947, 0.006

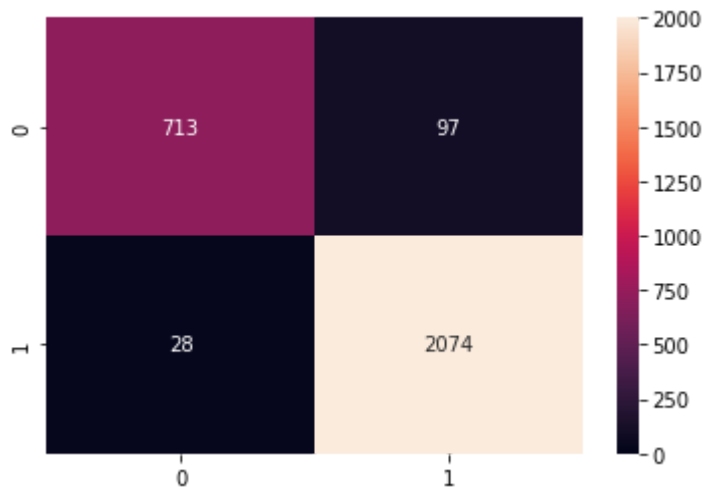
Rezultate test set

Rezultatele complete se regasesc in ***ptbdb_rforest_results.xlsx***

Best config results:

Metric name	Value(mean, std across classes)
Accuracy	0.956
Precision	0.962, 0.005
Recall	0.936, 0.052
F1	0.948, 0.024

Confusion matrix



Gradient boosted trees

Best configuration

n_estimators	max_depth	learning_rate
100	8	0.1

Rezultate cross validation

Rezultatele complete se regasesc in ***ptbdb_xgb_cv.xlsx***

Best config results:

Metric name	Value(mean, std across folds)
Accuracy	0.971, 0.002
Precision (macro)	0.969, 0.003
Recall (macro)	0.960, 0.004
F1 (macro)	0.964, 0.003

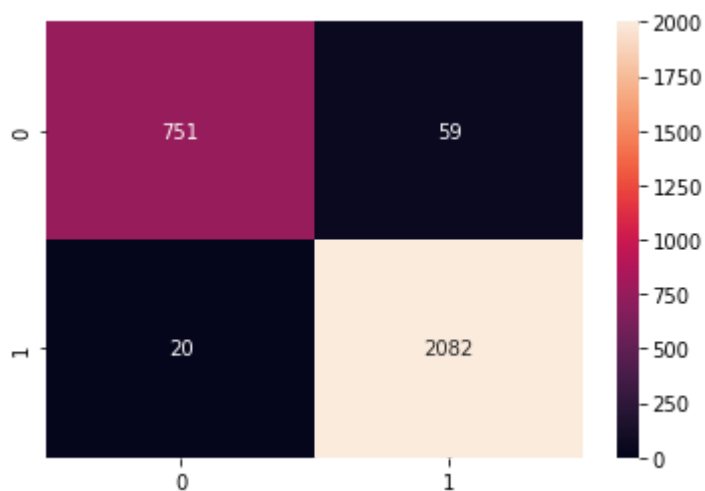
Rezultate test set

Rezultatele complete se regasesc in ***ptbdb_xgb_results.xlsx***

Best config results:

Metric name	Value(mean, std across classes)
Accuracy	0.973
Precision	0.973, 0.001
Recall	0.959, 0.032
F1	0.966, 0.016

Confusion matrix



Deep Learning

MLP

Best configuration (architecture + optimization setup)

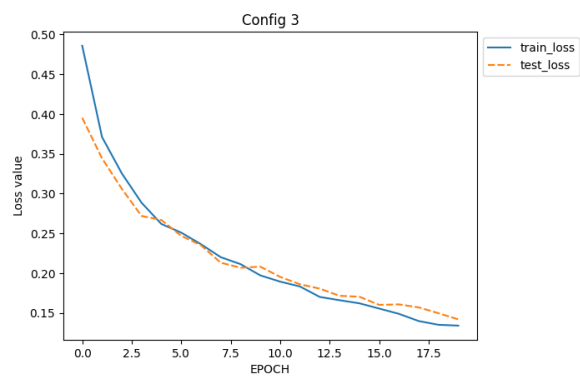
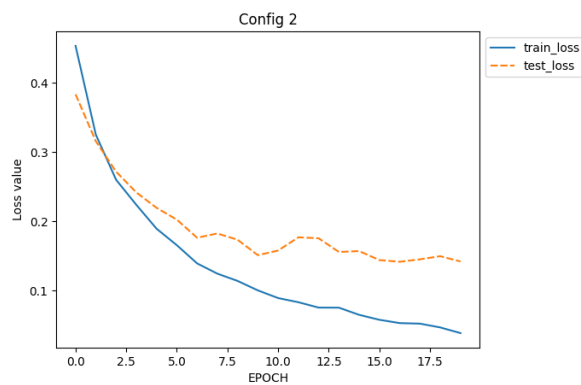
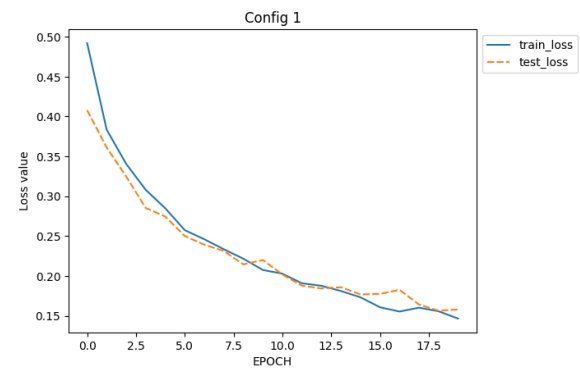
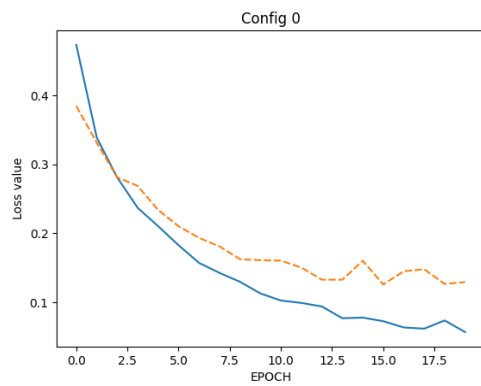
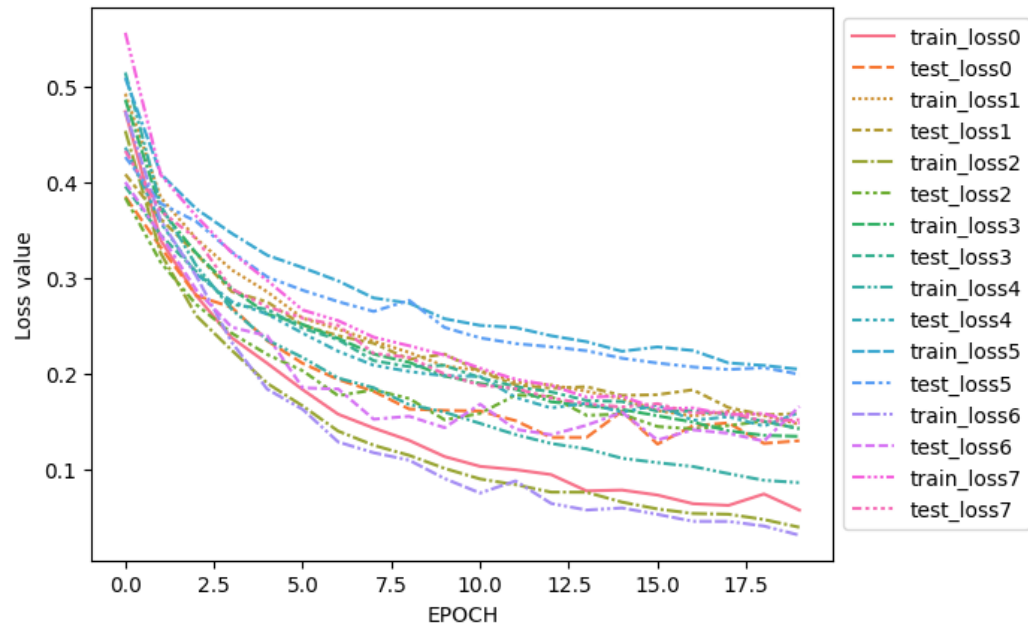
Parameter	Value
hidden_layers	2
n_units(hidden sizes from input to output)	[128, 64]
dropout	0.0
optimizer	Adam
learning rate	1e-3
batch size	256
epochs	20
scheduler	None
weight decay	None

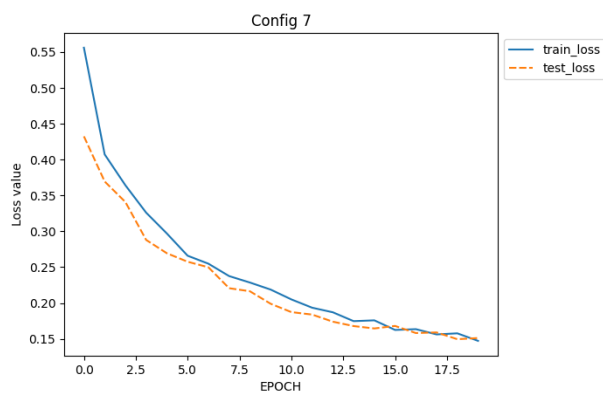
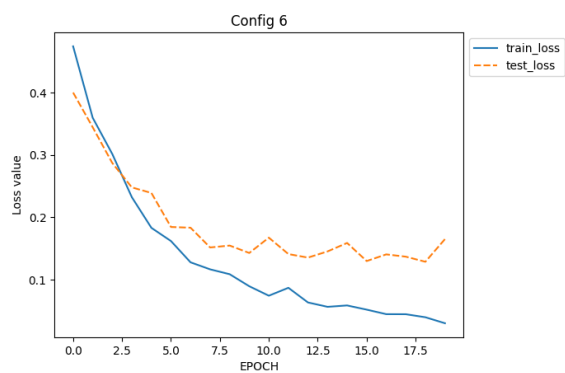
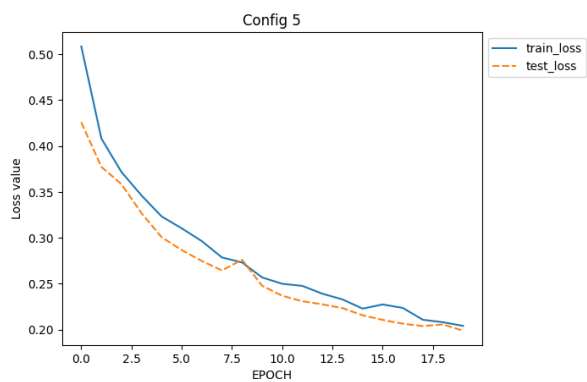
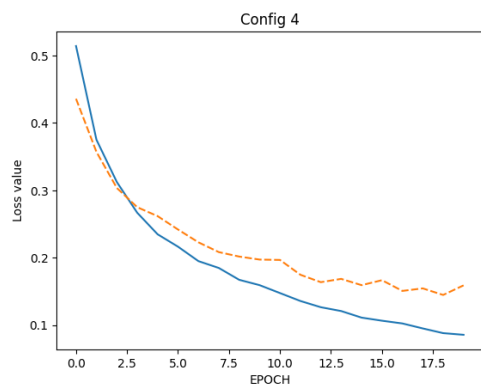
Rezultate best configuration

Metric name	Value
Accuracy	96.5
0_precision	93.8
0_recall	93.5
0_f1	93.6
1_precision	97.5
1_recall	97.6
1_f1	97.6

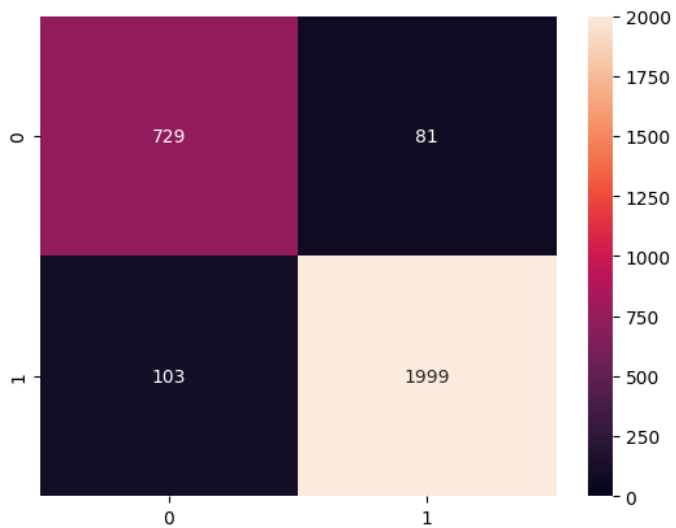
Loss evolution

Cele 8 configuratii corespund celor evidentiata din mlp_results.xlsx, fila PTBDB, in ordinea in care apar





Confusion matrix



Convolutional Neural Network

Best configuration (architecture + optimization setup)

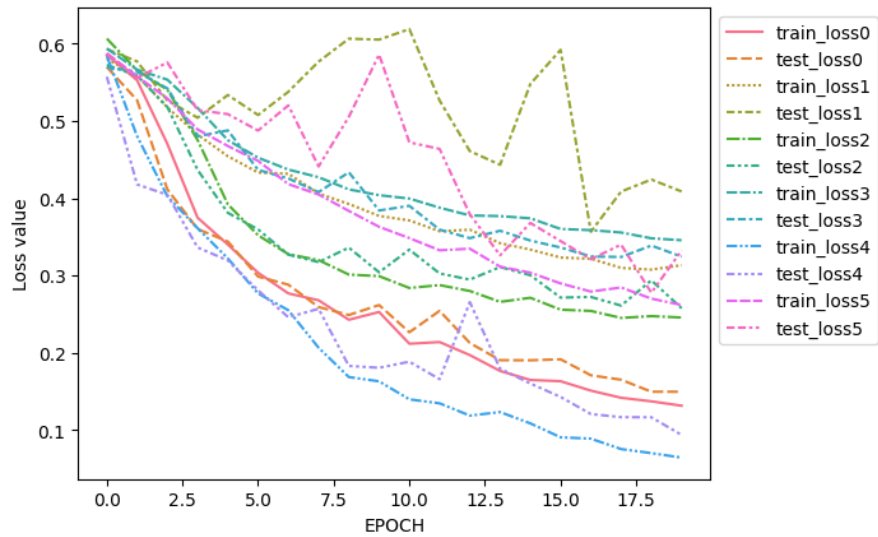
Parameter	Value
hidden_layers(CNN feature extractor part)	4
cnn_sizes	[32,64,128,256]
MLP classifier	[64,16], no dropout
dropout	0.0
optimizer	Adam
learning rate	1e-3
batch size	128
epochs	20
scheduler	None
weight decay	None

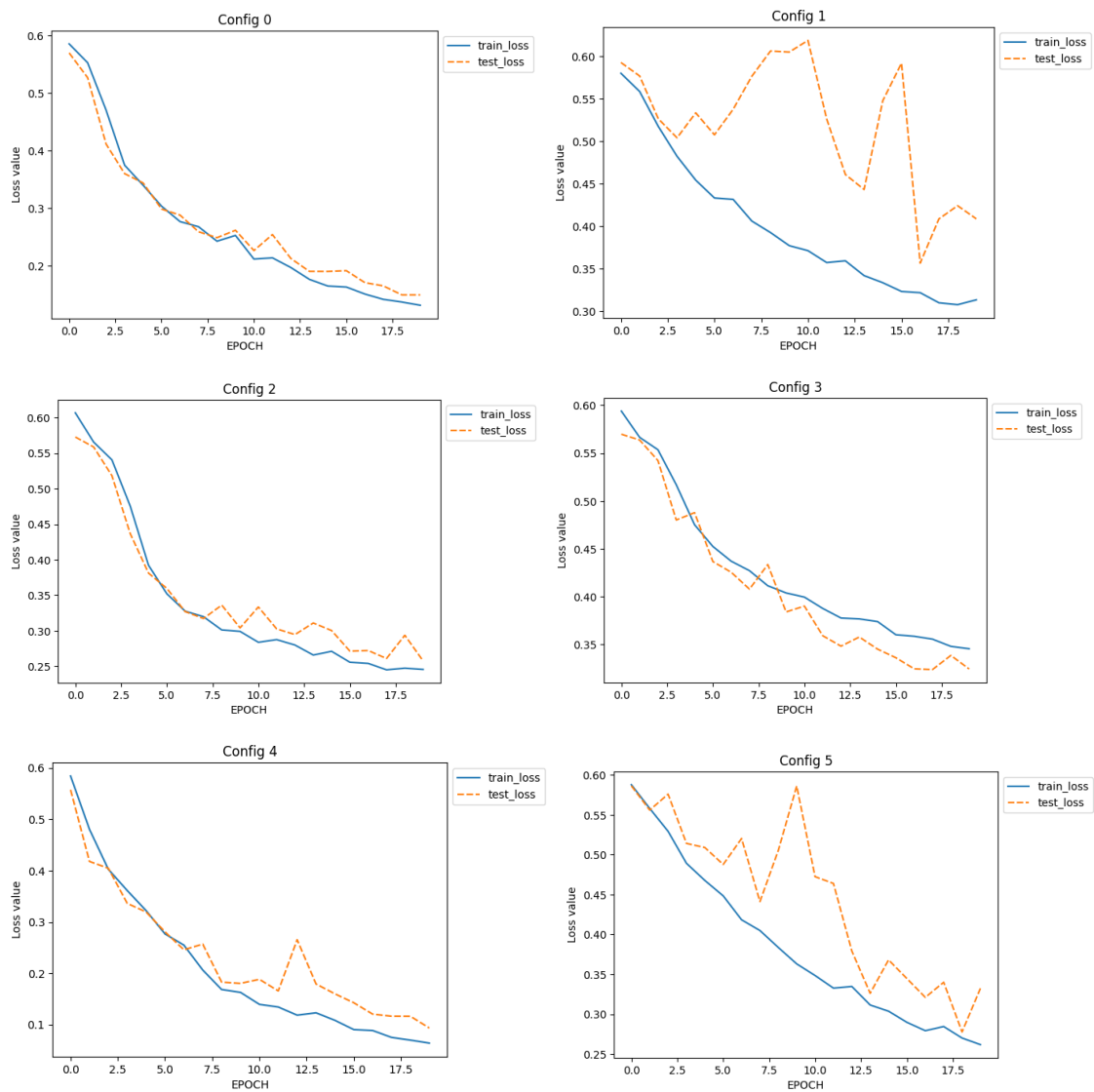
Rezultate best configuration

Metric name	Value
Accuracy	97.3
0_precision	99.1
0_recall	91.2
0_f1	95
1_precision	96.7
1_recall	99.7
1_f1	98.2

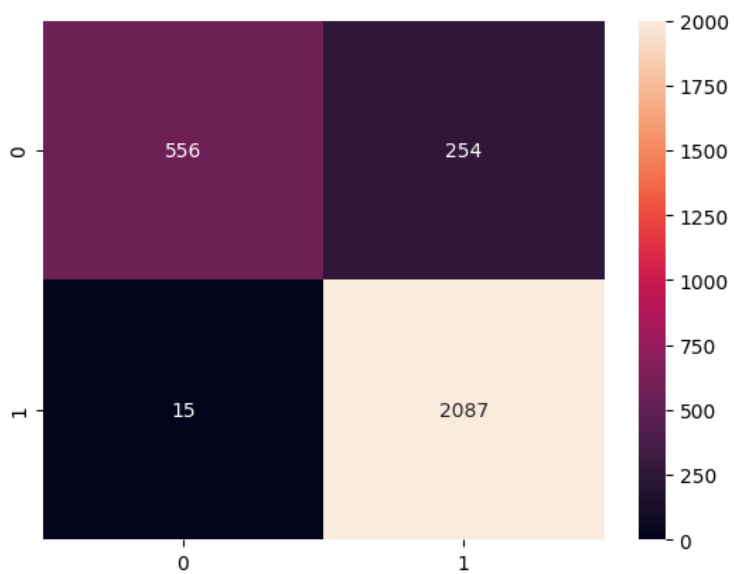
Loss evolution

Cele 6 configuratii corespund celor evidentiate din conv_results.xlsx, fila PTBDB, in ordinea in care apar





Confusion matrix



Recurrent Neural Network

Best configuration (architecture + optimization setup)

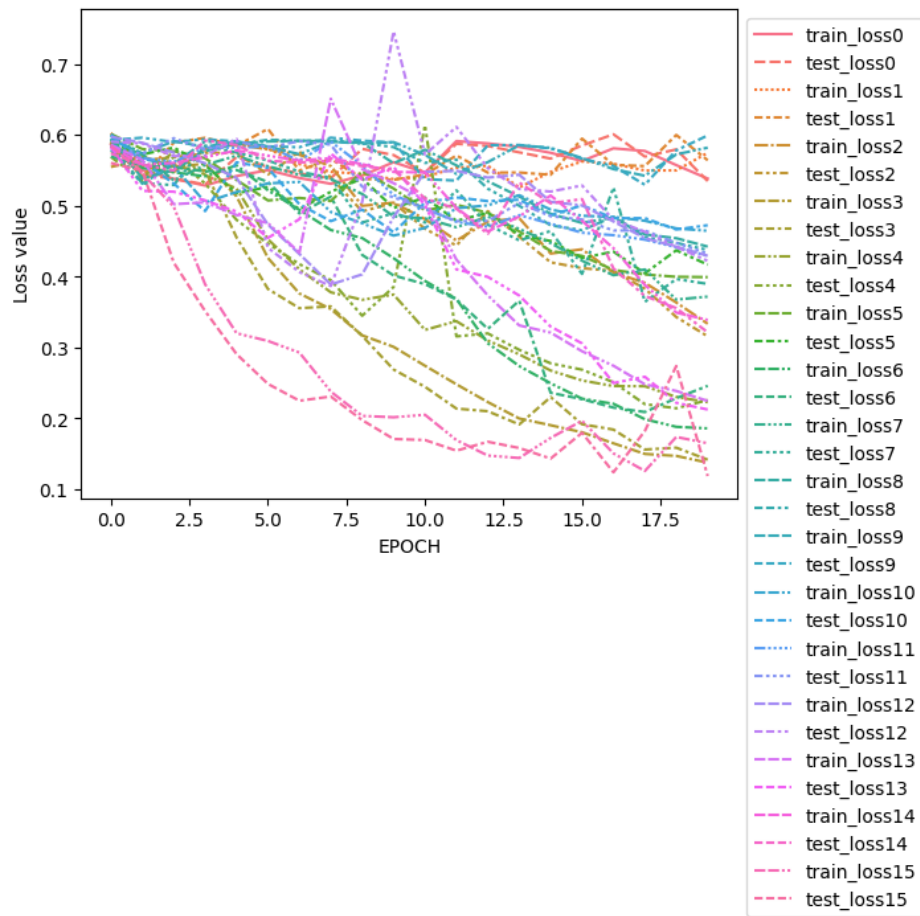
Parameter	Value
hidden_size	128
bidirectional	True
num_layers	2
dropout	0.3
MLP classifier	[64,16], no dropout
optimizer	Adam
learning rate	1e-3
batch size	32
epochs	20
scheduler	None
weight decay	None
other	Mean pooling of every step's output states passed further to MLP

Rezultate best configuration

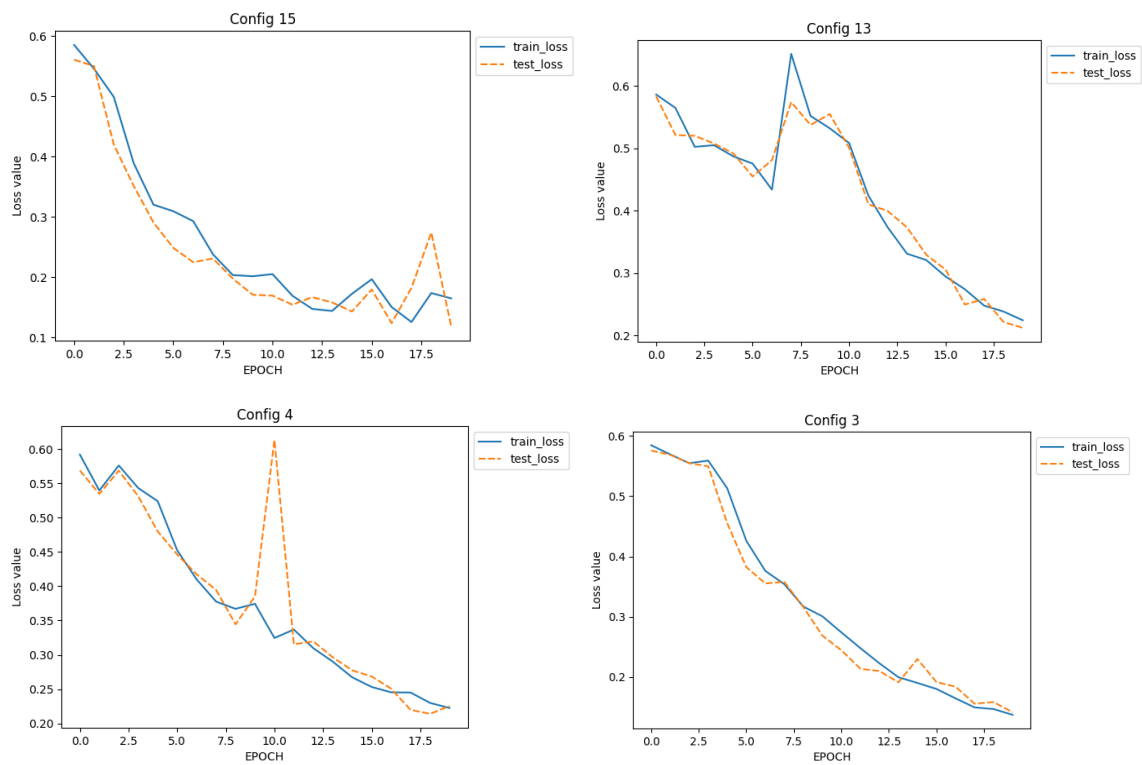
Metric name	Value
Accuracy	96.3
0_precision	93.8
0_recall	92.8
0_f1	93.3
1_precision	97.3
1_recall	97.6
1_f1	97.4

Loss evolution

Cele 16 configuratii corespund celor evidentiata din lstm_results.xlsx, fila PTBDB, in ordinea in care apar



Cele mai bune 4 configuratii au avut urmatoarea evolutie:



Confusion matrix

