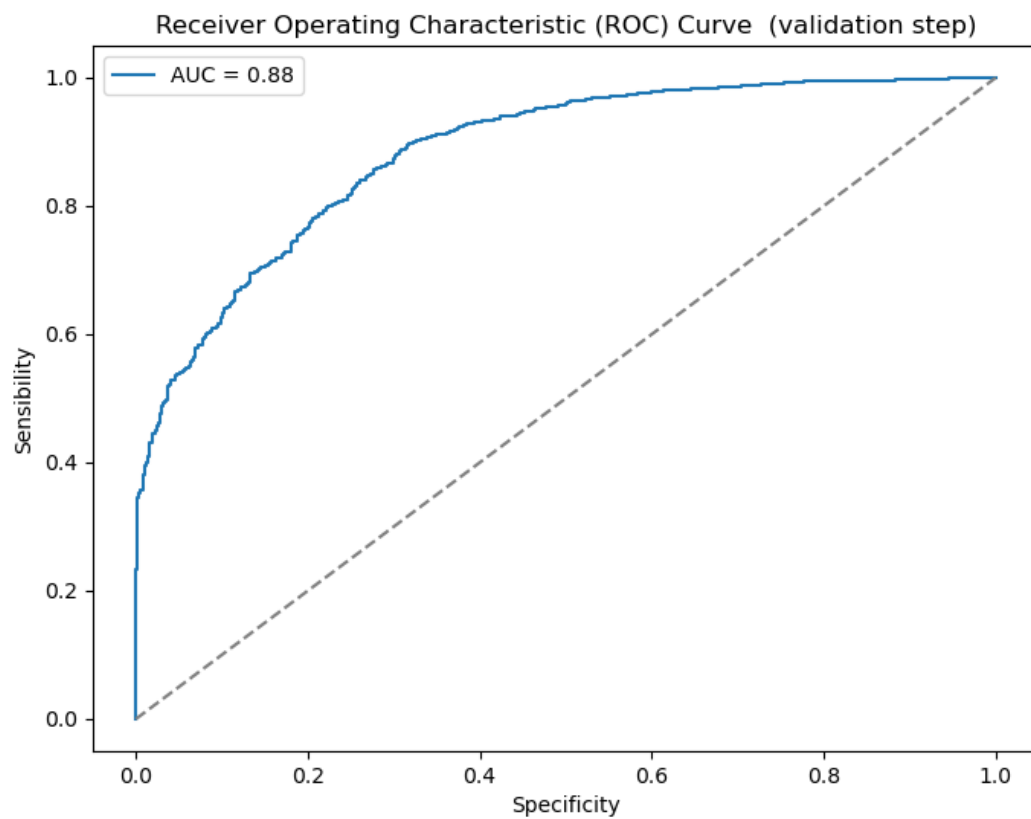
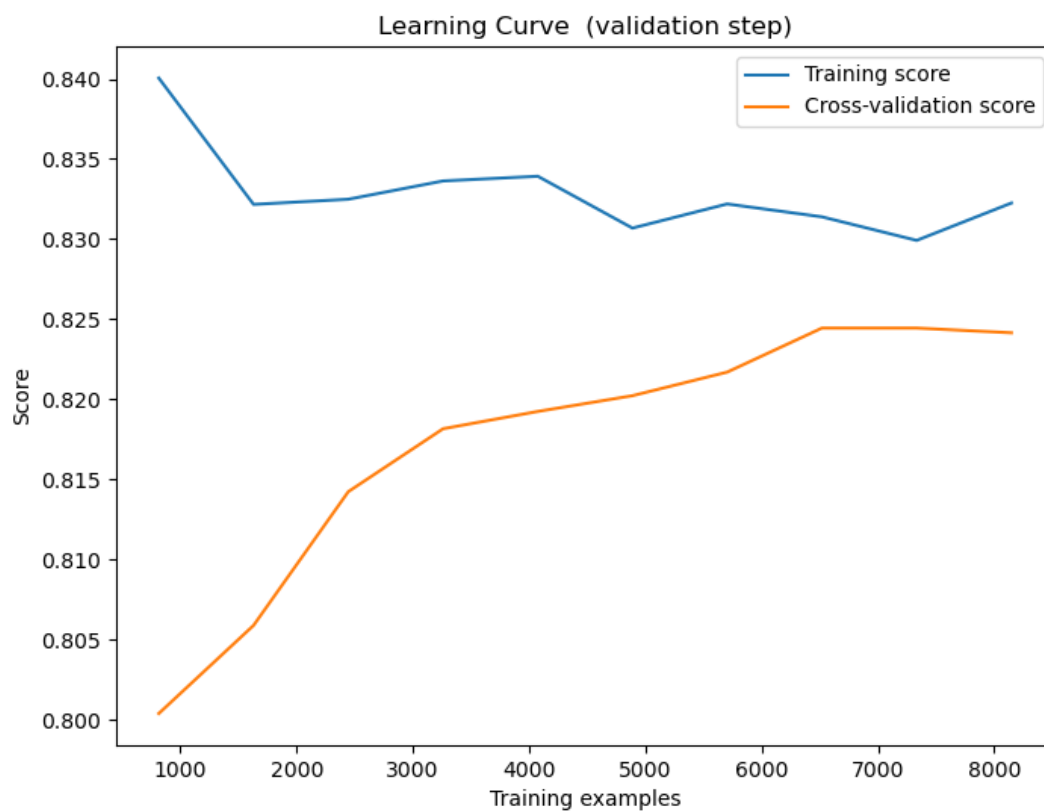
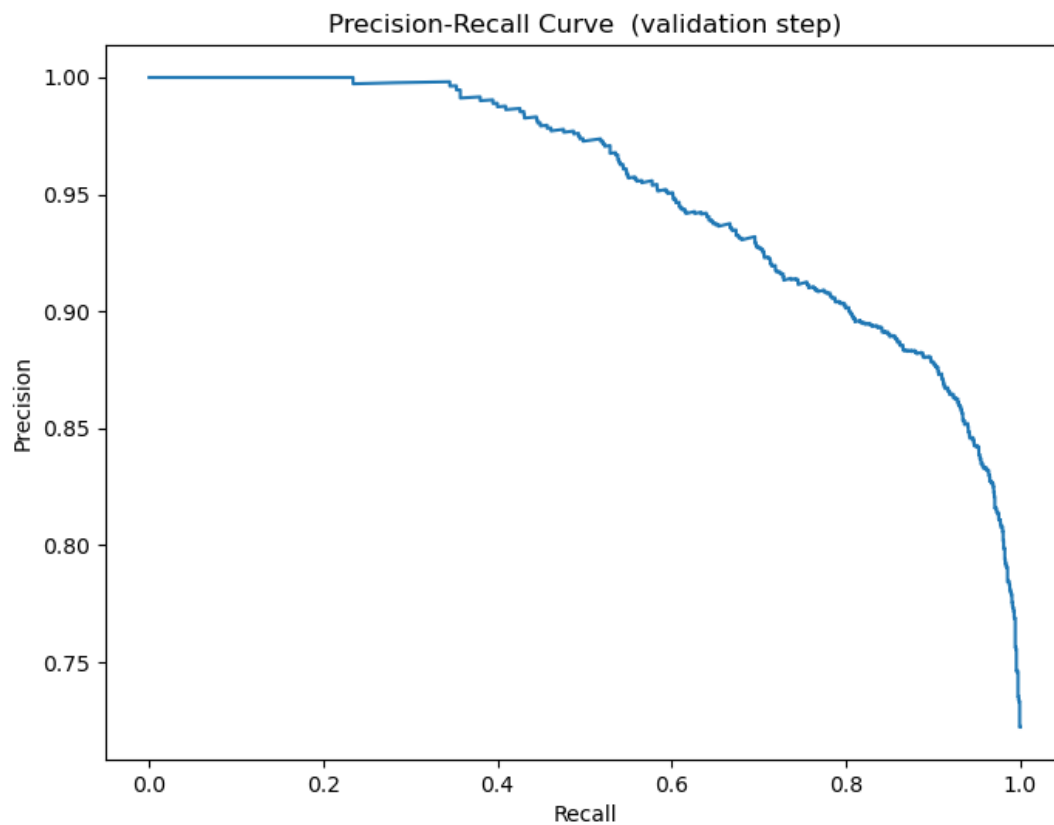


Graficul de mai sus arata cat de mult a contat in antrenare fiecare coloana a fisierului, fiind de asteptat importanta redusa a coloanelor de la 160 la final deoarece multe semnale din fisier nu acopera portiunea respectiva, fiind completate cu zerouri pentru a fi uniforma dimensiunea. Inca nu am gasit o explicatie pentru importanta redusa a coloanelor 25, 35 si 100.

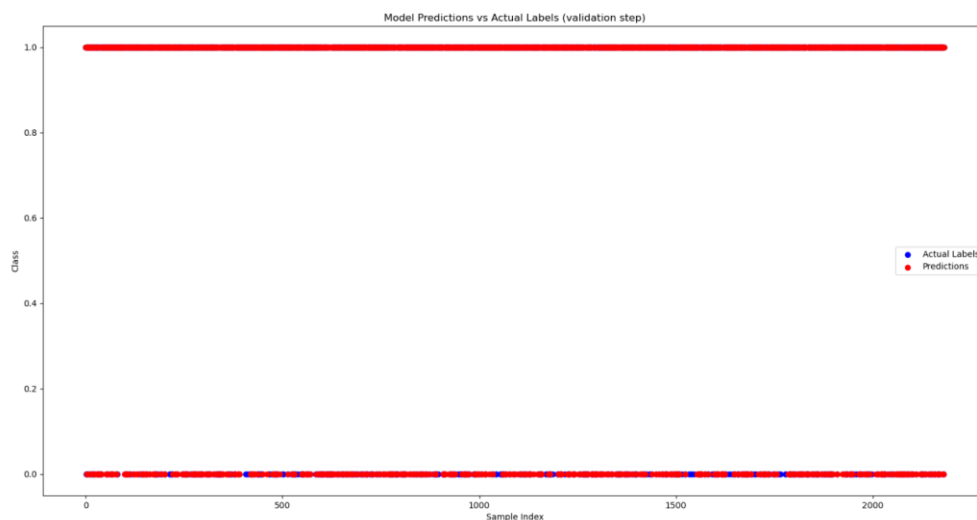


Deoarece AUC este aproape de 90% consider ca clasificatorul are performante bune.

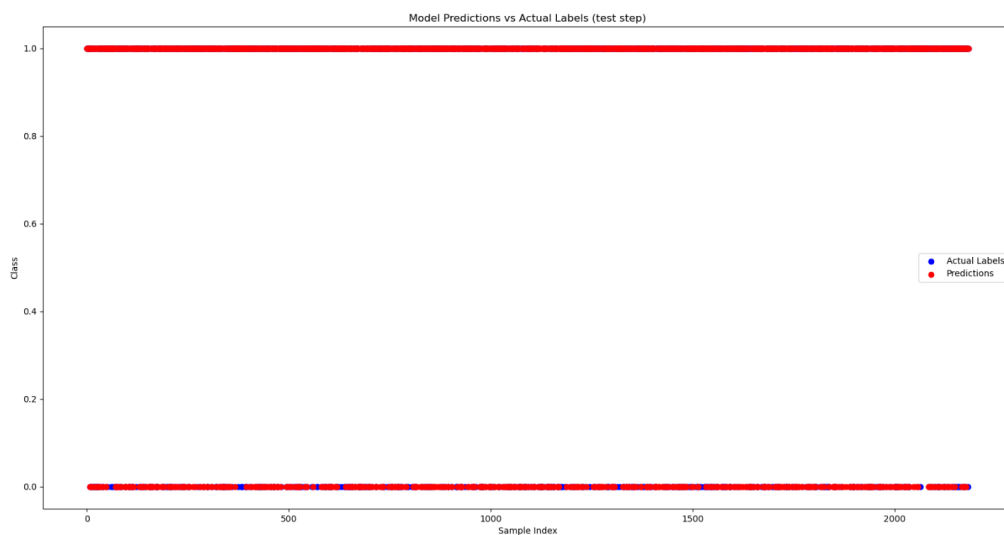
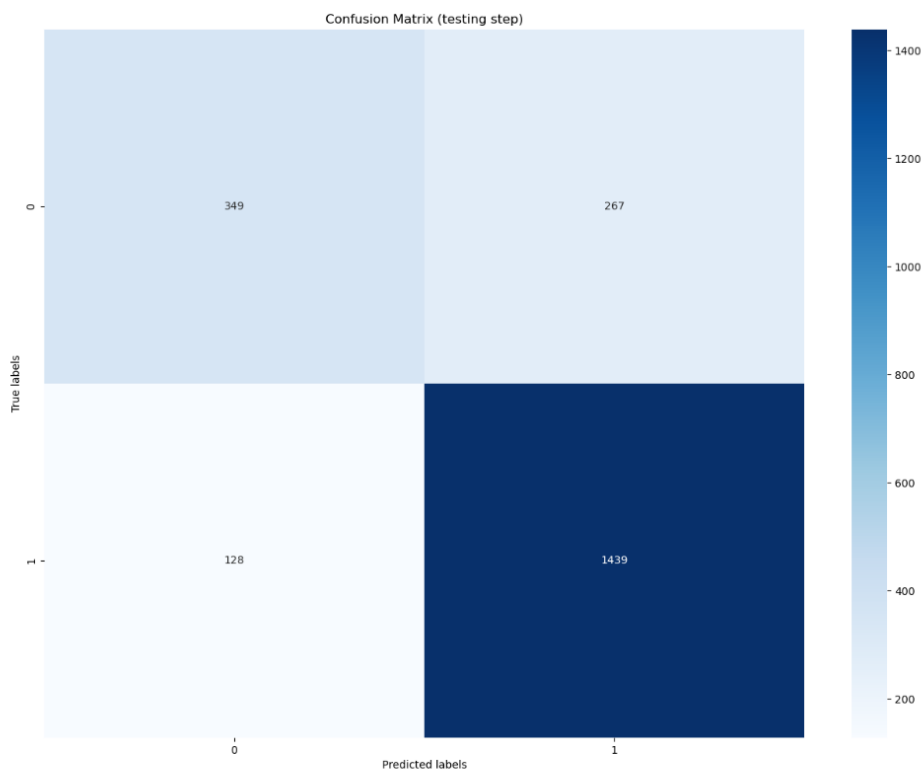




Aria de sub curba Precision-Recall este destul de mare, ceea ce confirma performantele clasificatorului.



Graficul de mai sus arata rezultatele grafice ale clasificarii, punctele albastre vizibile fiind clasificările gresite ale clasificatorului.



Mai sus avem un exemplu de rezultat pe setul de testare din datele initiale.

Programul permite si optiunea introducerii de noi date si clasificarea acestora, in cazul exemplului din cod acestea avand lable-uri si se poate verifica corectitudinea datelor.

Dupa cum se poate observa acuratetea nu este foarte mare, caz ce reflecta probabil faptul ca setul de date nu este ideal, dar nu pot spune cu certitudine deoarece nu sunt specialist in domeniul cardiologiei. Avem insa avantajul ca acest algoritm are un timp de executie mic (~7 secunde) si ar putea usura munca medicilor care trebuie sa verifice toate semnalele manual, si sa ii ajute sa filtreze semnalele sanatoase.

#### Bibliografie:

Setul de date: <https://www.kaggle.com/datasets/shayanfazeli/heartbeat>

Librarii folosite: pandas, sklearn, matplotlib, numpy, seaborn