

Tema2 IA

Manea Lidia-Elena - 341C4

12. Januar 2025

0.1 Cerinta 1

Am construit un MLP cu 4 straturi de fully connected. Primul strat primește 34 de intrări, iar numărul de neuroni este 256. Al doilea strat primește 256 intrări, iar numărul de neuroni este 128. Al treilea strat primește 128 intrări, iar numărul de neuroni este 64. Al patrulea strat primește 64 intrări, iar ieșirile sunt numărul de clase (10 sau 70 în funcție de setul de date). Ca funcții de activare, am folosit numai RELU, după primele 3 straturi. Pentru antrenare, am folosit optimizerul Adam cu lr de $\text{pow}(10, -5)$ și funcția de eroare Cross Entropy Loss. Numărul de epoci este de 100 pentru ambele seturi de date, iar dimensiunea batch ului este de 64. Rezultatele obținute (0.85 acuratețe pentru fructe și mnist) sunt mai bune decât cele pentru Logistic Regression și xgboost, dar mai slabe decât cele pentru SVM. Modelul putea fi îmbunătățit prin finetuning.

0.2 Cerinta 2

Pentru MLP ul peste imagini, am folosit doar 2 straturi liniare. Primul strat primește dimensiunea imaginii ca input și outputul este dat de 128 neuroni. Funcția de activare folosită este RELU. Al doilea strat primește 128 neuroni la intrare și la ieșire numărul de clase specific seturilor de date. Pentru antrenare, am folosit optimizerul Adam cu lr de $\text{pow}(10, -5)$ și funcția de eroare Cross Entropy Loss. Pentru ambele seturi de date am folosit 66 de epoci, iar dimensiunea batch ului este de 64. MLP ul pe imagini este mai performant decât cel initial (are acuratețe de 0.87 pe ambele seturi de date), deși modelul este mult mai simplu. Asadar, a reușit să surprindă detaliile. În continuare, nu este cel mai performant model, însă este mai bun decât xgboost și logisticregression.

0.3 Cerinta 3.1

Am creat un deepconvnet care are straturi de convoluție, batchnorm, avgpool, linear și adaptiveavgpool. Straturile sunt în următoarea ordine: conv2d cu input 3 și out 16, BatchNorm, funcția RELU, AvgPool, conv2d cu input 16 și output 64, BatchNorm, funcția RELU, AvgPool, conv2d cu input 64 și output 256, BatchNorm, RELU, AvgPool, AdaptiveAvgPool, flatten și 2 linears cu input 256 și output 128, respectiv input 128 și output numărul de clase. Pentru antrenare, am folosit optimizerul Adam cu lr de $\text{pow}(10, -5)$ și funcția de eroare Cross Entropy Loss. Pentru ambele seturi de date am folosit 35 de epoci, iar dimensiunea batch ului este de 64. Acesta este un model bun, care obține rezultate bune pentru ambele seturi de date (0.93 pentru fructe și 0.87 pentru mnist).

0.4 Cerinta 3.2

Am pastrat acelasi pattern ca la cerinta 3.1, exact acelasi model, aceiasi parametri, nu am schimbat nimic in afara de transformarile adaugate in plus, pentru ca masuratoarea sa fie una coreta. Rezultatele pe test au fost mai mici pe ambele seturi de date, pentru fructe fiind 0.88 si pentru mnist fiind 0.8.

0.5 Cerinta 4

Am folosit reseaua resnet 18 preantrenata. Am modificat ultimul strat ca outputul sa fie dat de numarul de clase pentru fiecare set de date. Pentru antrenare, am folosit optimizerul SGD cu lr egal cu $\text{pow}(10, -4)$, iar functia de loss a ramas in continuare CrossEntropyLoss. De asemenea, dimensiunea batch-ului este de 64, iar numarul de epoci este de 25 pentru fructe si 35 pentru mnist. Am obtinut acelasi rezultate ca la cnn.

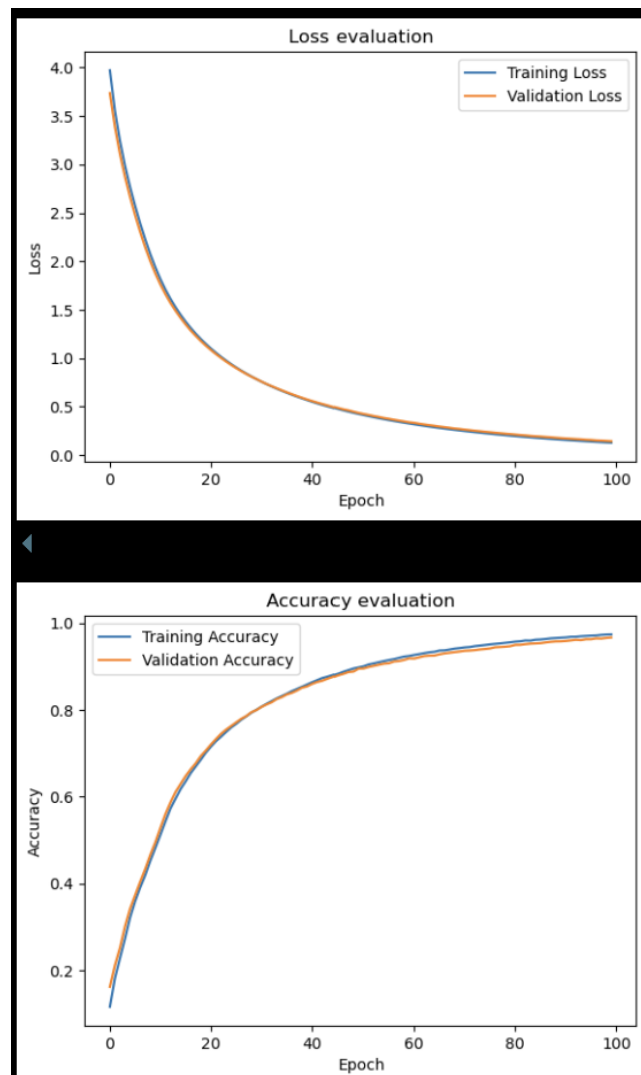


Abbildung 1: Cerinta 1 Fruits

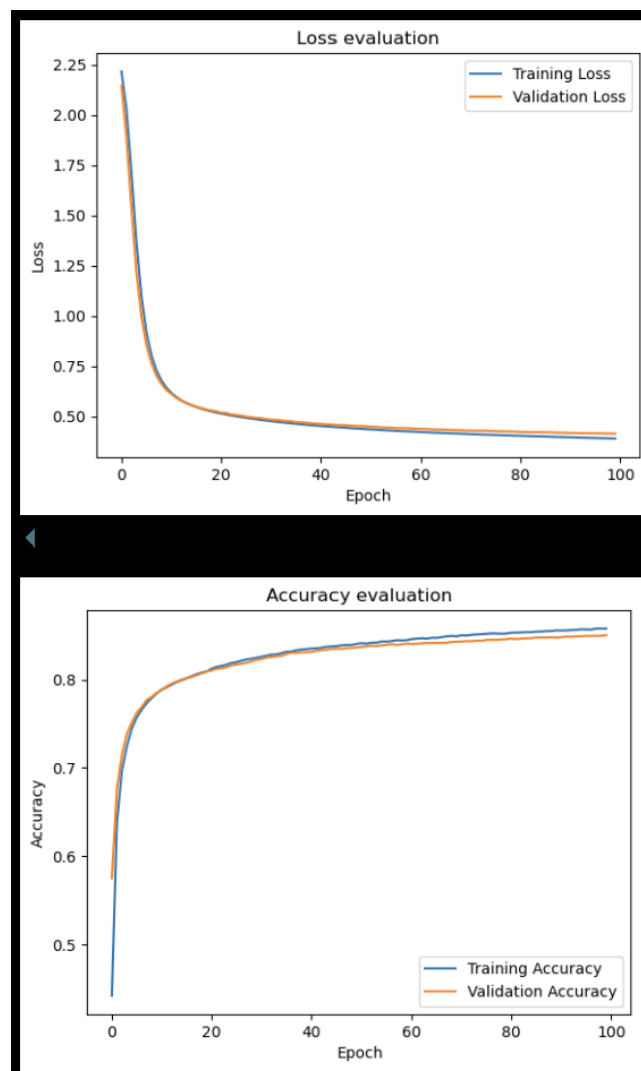


Abbildung 2: Cerinta 1 mnist

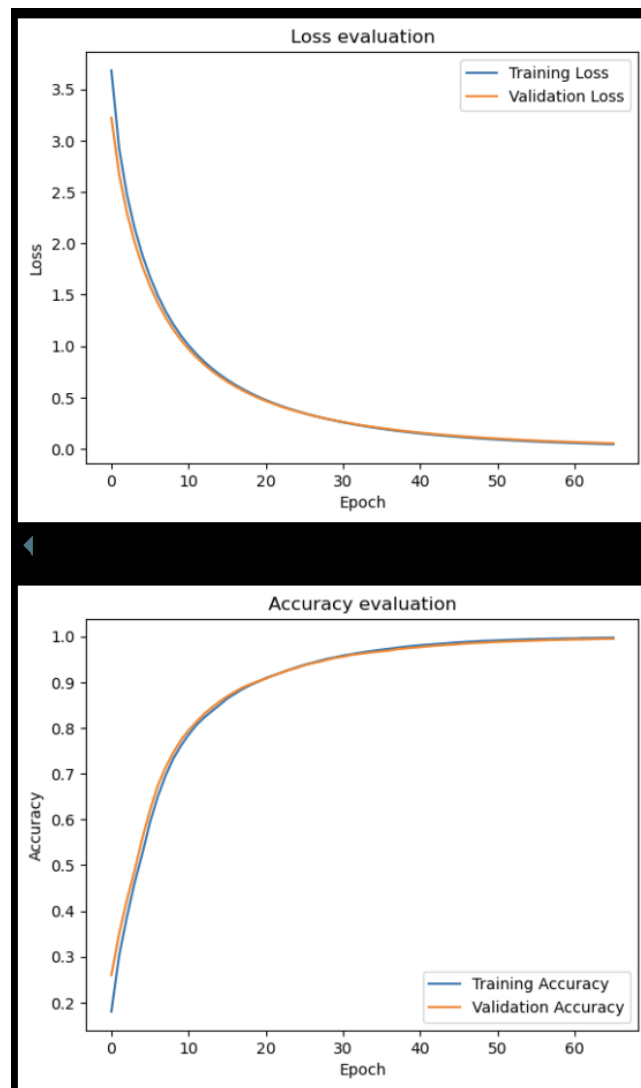


Abbildung 3: Cerinta2 fruits

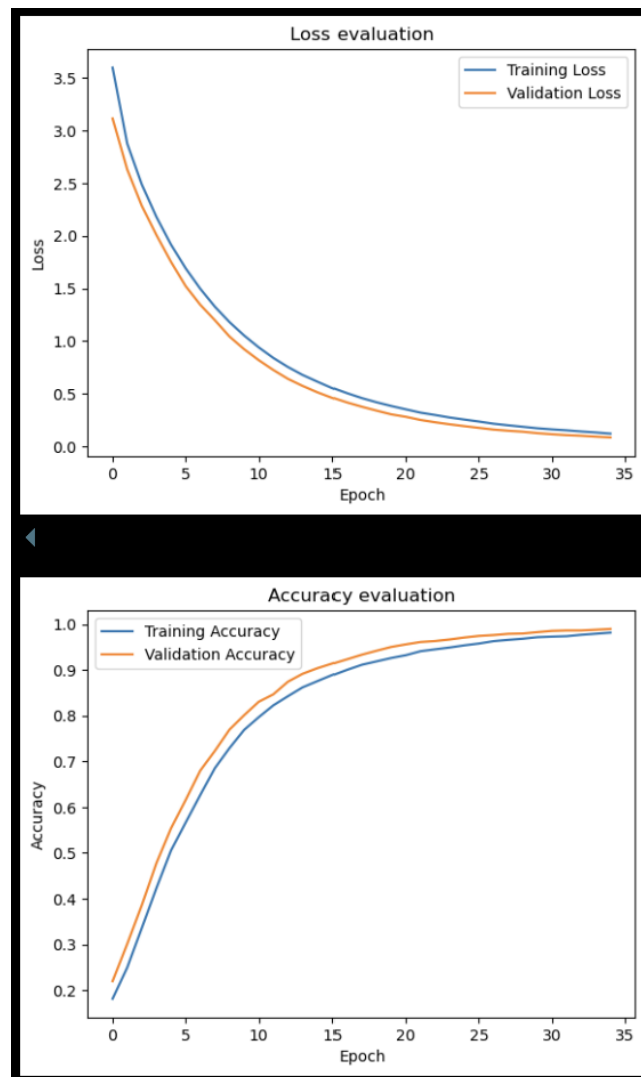


Abbildung 4: Cerinta3.1 fruits

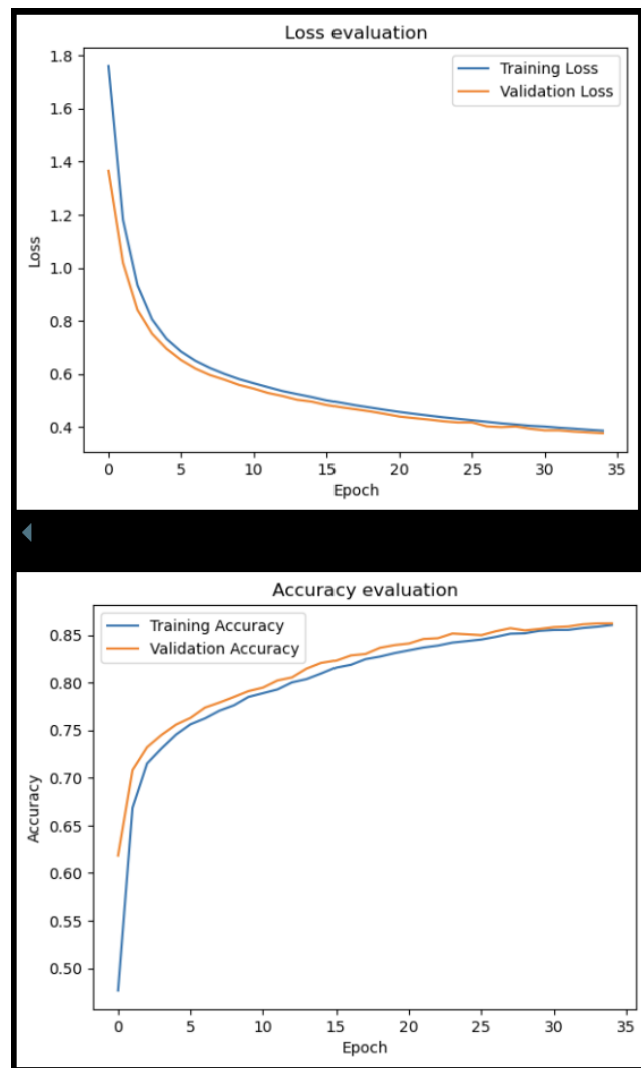


Abbildung 5: Cerinta3.1 mnist

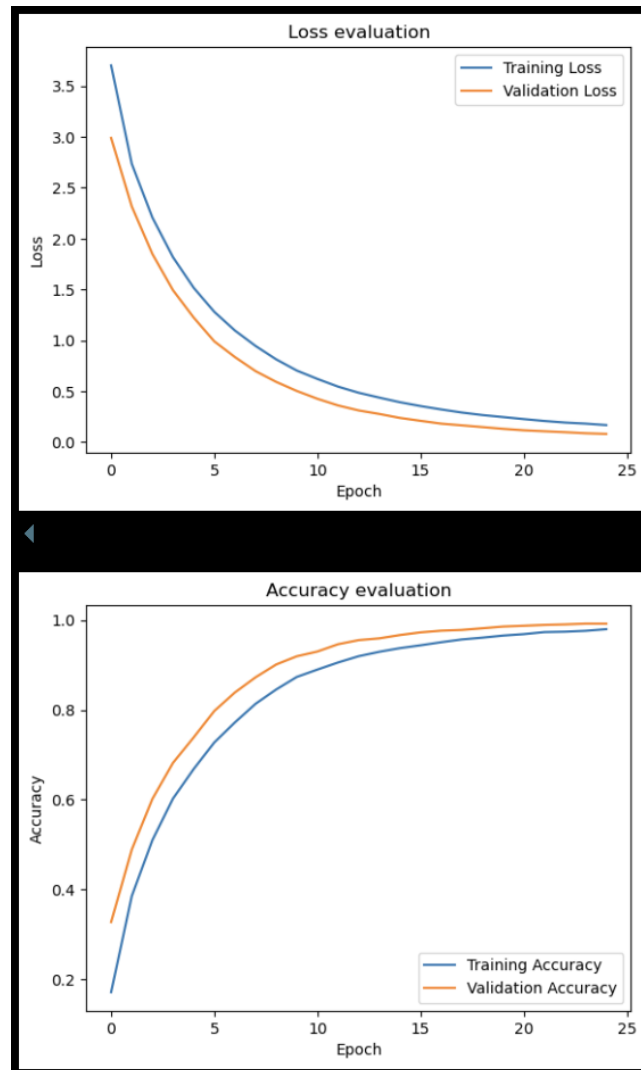


Abbildung 6: Cerinta4 fruits

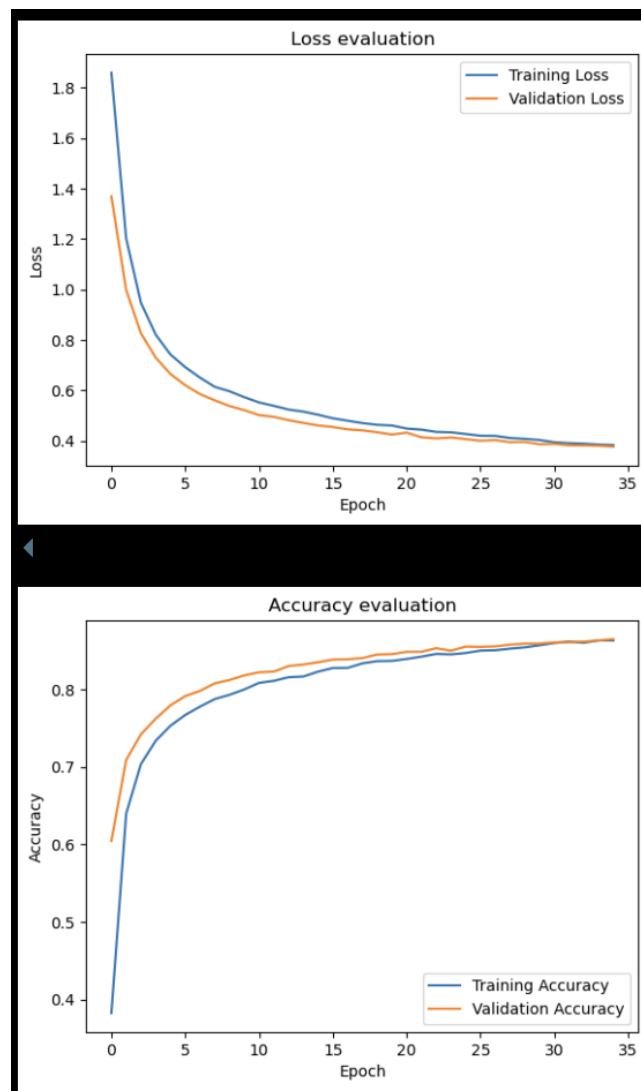


Abbildung 7: Cerinta4 mnist

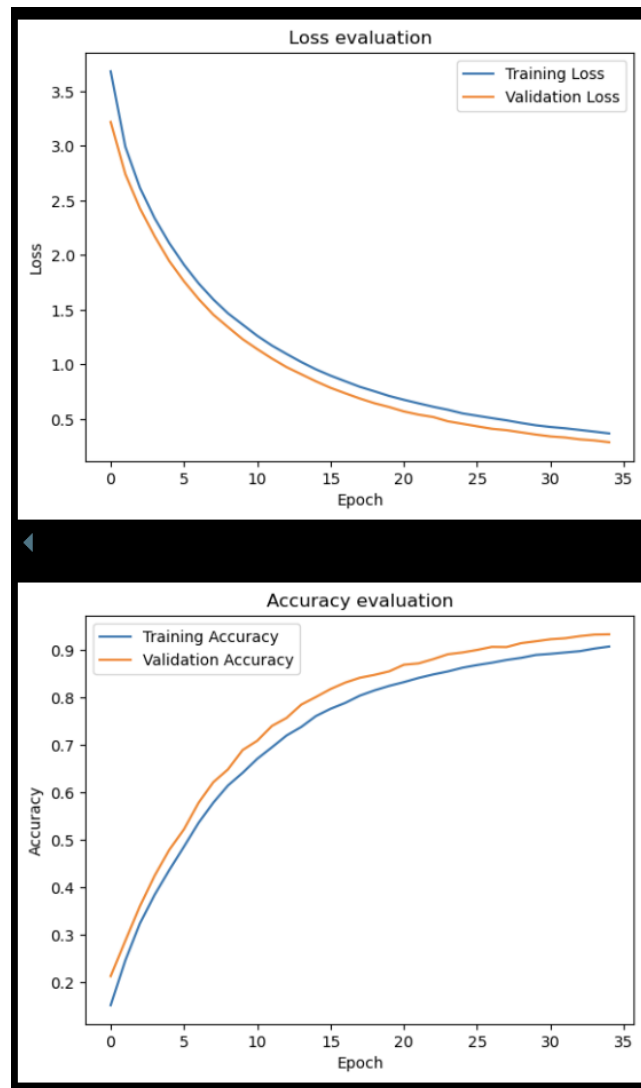


Abbildung 8: Cerinta3.2 fruits

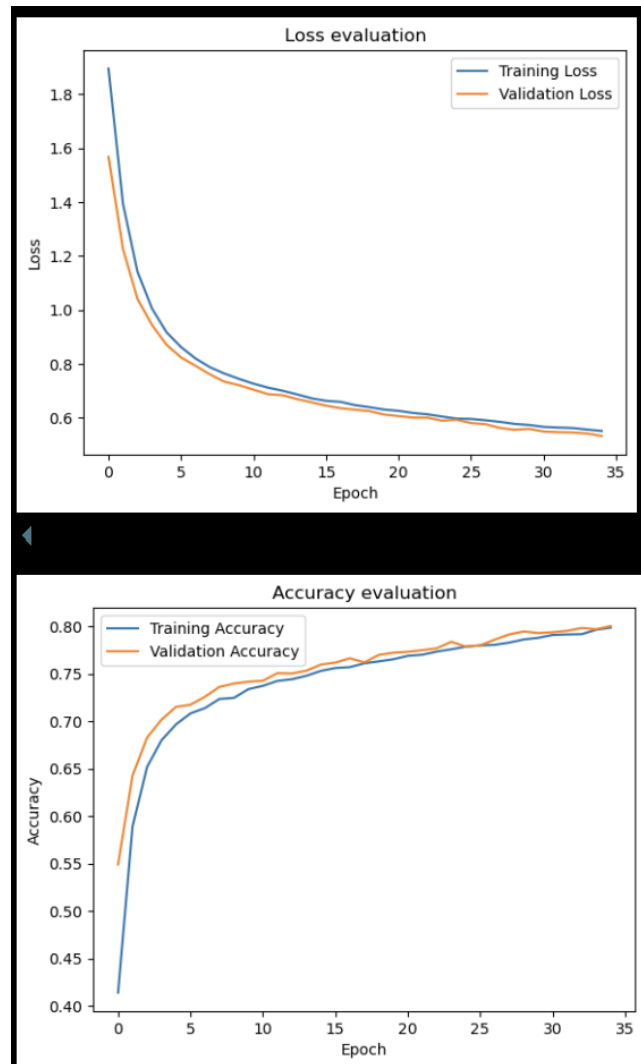


Abbildung 9: Cerinta3.2 mnist