Maskininlärning

Kunskapskontroll 2

Lidiia Kashevarova

EC Utbildning

Examensarbete

2024 Mars

Abstract

This work is the final test of knowledge in the “Machine Learning” course. The work includes answers to theoretical questions of the machine learning course and a practical part. In the practical part, I used MNIST data to create three models. These three models were trained and I chose the best models with the help of validation. Based on the results of models with validation MNIST-data, two models showed an accuracy of more than 90% (Random Forest - 95.6%, K-nearest neighbors - 92.9) and with test MNIST-data (Random Forest - 95.7%, K-nearest neighbors -94.7%). I also tested these two models with photographs that I took on my mobile phone. In this case K-nearest neighbors shows the better result than Random Forest. In addition, the Streamlit application was created. This application can read images and make predictions with numbers using a web camera or downloading previously saved images. I used K-Nearest Neighbors to create a Streamlit application.

1. Inledning…… 4

2. Teori……
5

2.1. Confusion Matrix…… 5

2.2. Valideringsdata …… 6

2.3. Logistisk regression model…… 7

2.4. Random Forest model…… 7

2.5. K-nearest neighbors (KNN)-modellen …… 7

3. Metod ……
8

4. Resultat och Diskussio ……... 9

4.1. Träning av modeller och definition

precision av förutsägelse….. ……. 9

4.2. Koden för mina egna bilder från

mobiltelefon…… 9

4.3. Streamlit application…… 10

5. Slutsatser……
12

6. Teoretiska frågor…… 13

7. Självutvärdering…… 17

8. Källförteckning…… 18

1. Inledning

Studiet av maskininlärning är viktigt och relevant. Redan idag används maskininlärning inom många områden, till exempel i självkörande bilar eller ansiktsigenkänning. Maskininlärning erbjuder verktyg och tekniker för att effektivt utföra många databearbetningsuppgifter. Därför kan maskininlärning hjälpa till att förbättra beslutsfattandet genom att tillhandahålla värdefulla insikter och datadrivna förutsägelser, vilket kan leda till förbättrade affärsstrategier och resultat (Azure, Internet) .

Huvudmålet med detta arbete är att visa på de kunskaper jag tillägnat mig under kursen i Machine Learning, för att uppfylla syftet:

- jag svarade på teoretiska frågor;

- jag skapade 3 modeller (Logistisk regression model, Random Forest model, K-nearest neighbors (KNN)-modellen);

- jag tränade dessa modeller på data MNIST;

- med hjälp av valideringsdata valde jag Random Forest och K-nearest neighbors modeller som har högsta prediktionsnoggrannhet;

- jag testade Random Forest och K-nearest neighbors modellerna på testdatan;

- jag testade Random Forest och K-nearest neighbors modellerna på fotografier som jag har tagit och bearbetat själv;

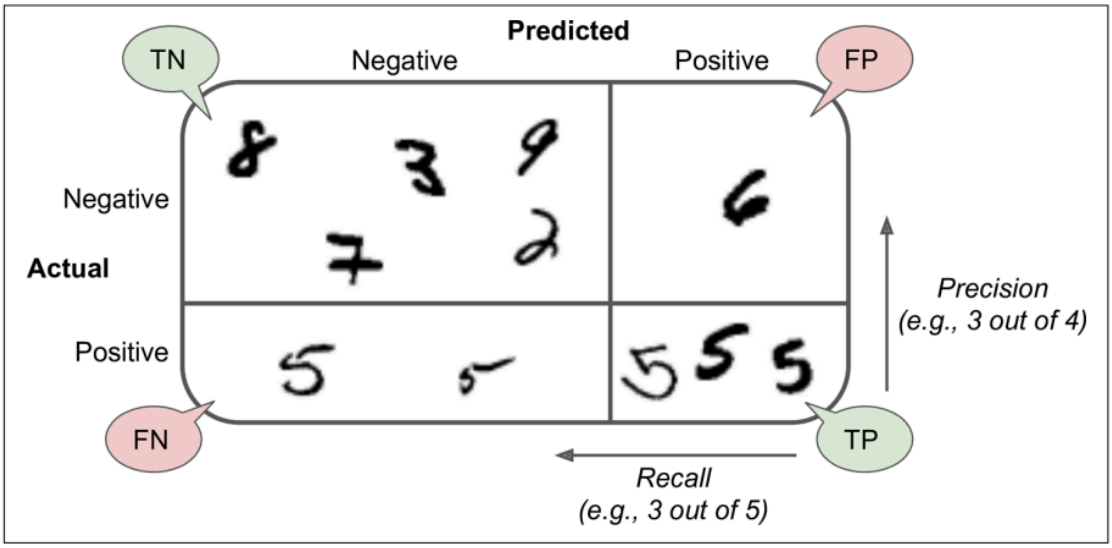
- jag skapade en Streamlit applikation som kan ta eller ladda upp en bild och prediktera siffror som står på bilden.

Detta arbete är viktigt för mig personligen eftersom det gjorde det möjligt för mig att strukturera teoretiska kunskaper, att träna på att skapa och testa modeller inte bara på välbearbetade MNIST-testdata utan också på riktiga fotografier, förbearbeta bilder så att modellen kan läsa dem och göra rätt förutsägelse.

2. Teori

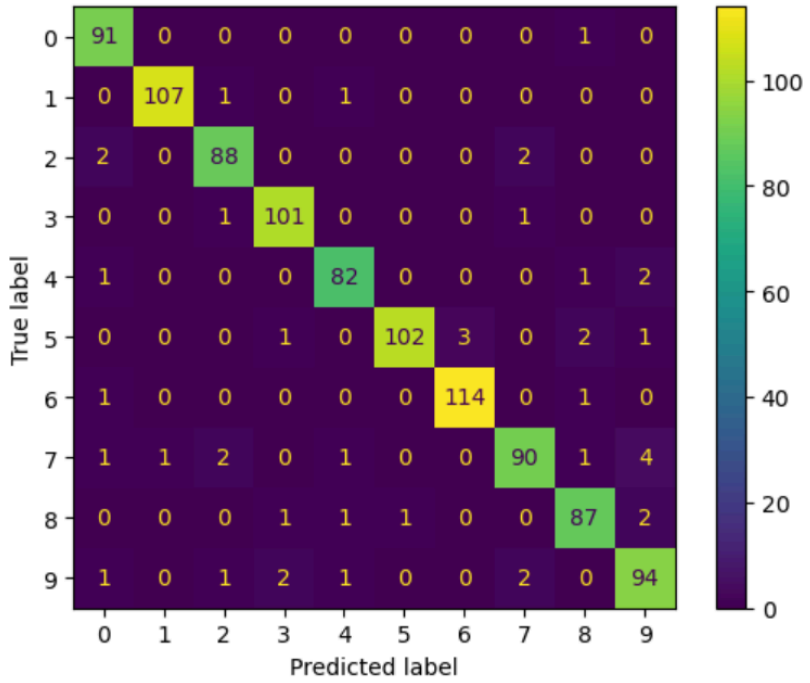
2.1. Confusion Matrix

För att utvärdera träningsresultaten på de 3 modellerna använde jag Confusion Matrix . I boken Hands-on Machine Learning with Scikit-Learn, Keras & TensorFlow beskrivs huvud tanken av Confusion Matrix som "The general idea is to count the number of times instances of class A are classified as class B. (Géron s. 9) De där faktisk kan vi se i Confusion Matrix , till exempel, hur många gånger modellen identifierade 3 som 3, dvs. gav rätt svar och hur många gånger definierade modellen 3 som 5 d.v.s. gav fel svar.



Fugur 1. En illustrerad av Confusion Matrix (Källa: Hands-on Machine Learning with Scikit-Learn, Keras & TensorFlow, Aurélien Géron s. 92)

```
display_confusion_matrix(y_val, y_val_pred_random_forest)
```



Figur 2. The result av validation of the Random forest model(Jupiter notebook, code Model_selection)

För att beräkna Confusion Matrix behöver man ha de faktiska målen och får det förutsagda värdet som ett resultat av modellträning. För att få predikterade värden kan vi använda valideringsdata som inte användes i modellträning. Confusion Matrix hjälper oss att tydligt se var matrisen gör flest fel.

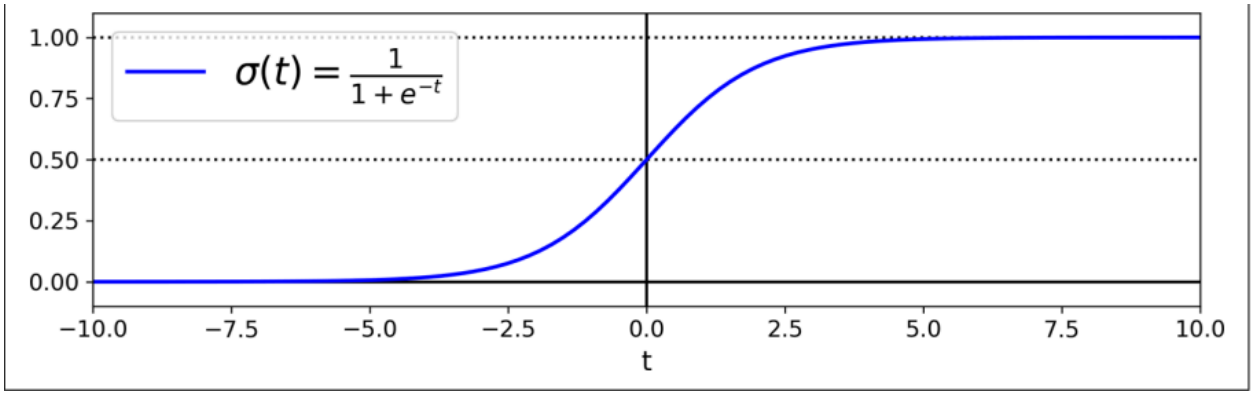
2.2. Valideringsdata

Grunden för att lyfta fram valideringsdata är behovet av att undvika att anpassa en modell till en specifik datamängd. I boken Hands-on Machine Learning with Scikit-Learn, Keras & TensorFlow beskrivs ett problem när man mäter ett gemensam fel på test setet flera gånger och anpassade modellen och hyperparametrarna för att skapa en bättre modell för exakt denna uppsättning. Författarna av boken rekommenderar “A common solution to this problem is called holdout validation: you simply hold out part of the training set to evaluate several candidate models and select the best one. The new held-out set is called the validation set (or sometimes the development set, or dev set). “ (Géron s.31)

2.3. Logistisk regression model

Logistisk regressionsmodell uppskattad sannolikhet (vektoriserad form):

$$\hat{p} = h_{\boldsymbol{\theta}}(\mathbf{x}) = \sigma(\mathbf{x}^{\mathsf{T}}\boldsymbol{\theta})$$



Figur 3. Logistik funktion (Källa: Hands-on Machine Learning with Scikit-Learn, Keras & TensorFlow s. 143)

2.4. Random Forest model

Random Forest är ett ensemble av beslutsträd i allmänhet. Algoritmen av Random Forest beskrivs i I boken Hands-on Machine Learning with Scikit-Learn, Keras & TensorFlow som "instead of searching for the very best feature when splitting a node , it searches for the best feature among a random subset of features" (Gérons.197)

2.5. K-nearest neighbors (KNN)-modellen

Algoritmen K-nearest neighbors (KNN) är en klassificerare som använder närhet för att klassificera eller förutsäga grupperingen av en enskild datapunkt. KNN kan användas för både regressions- och klassificeringsproblem, men ofta används den som en klassificeringsalgoritm utifrån antagandet att liknande punkter kan hittas nära varandra (IBM, Internet)

3. Metod

I mitt arbete använde jag MNIST-datan. De 25 000 observationerna valdes ut för modellträning, 1 000 för validering och 1 000 för testning. Därefter, för att testa modeller, använde jag fotografier med siffror som jag tog och förbearbetade själv. För att skapa en Streamlit applikation tränade jag modellen K-nearest neighbors separat på 50 000 observationer från MNIST-datan.

4. Resultat och Diskussion

4.1. Träning av modeller och definition noggrannheten av förutsägelser

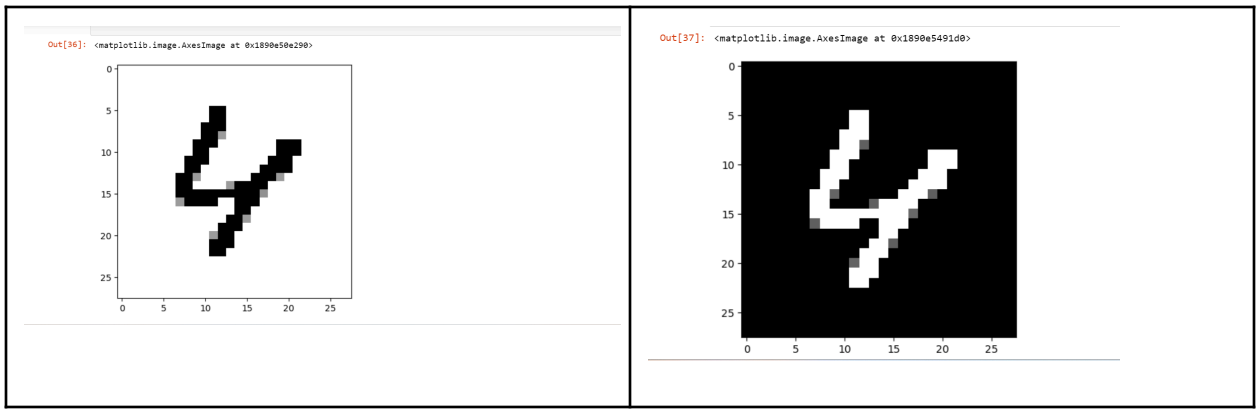
Precision för olika modeller	
Logistisk regression model	89,7%
Random Forest model	95,6%
K-nearest neighbors (KNN)-modellen	92,9%

Tabell 1: Precision för de tre valda modellerna.

Som ett resultat av validering av tre modeller visades det bästa resultatet av Random Forest. Denna modell gav korrekta svar 95,6 % av gångerna. Att testa modellen på testdata gav också ett högt resultat – 95,7 % av rätt svar.

4.2. Koden för mina egna bilder från mobiltelefon

För bilderna som jag själv förberedde visade sig Random Forest modellen helt oanvändbar. Jag antar att arbetet med den här modellen kräver fotografier av mycket hög kvalitet och komplex förbearbetning. För de fotografier som jag själv tog visade K-nearest neighbors (KNN)-modellen sig mest lämpliga. Men även denna modell inte visade samma precision som under testningen.



Resultat av bearbetning av en bild och en förutsägelse: *Figur 4. Ett foto som var bearbetade med hjälp av Open Source Computer Vision Library och förändringar pixelfärg*

```
In [38]: # Prediction av KNN-model on the preprocessed image

         prediction_knn = grid_search_knn.predict(flattened_image)
         print("Prediction knn:", prediction_knn)

         Prediction knn: [4]

In [39]: # Prediction av Random Forest model on the preprocessed image

         prediction_rf = random_forest.predict(flattened_image)

         print("Prediction Random Forest:", prediction_rf)

         Prediction Random Forest: [8]
```

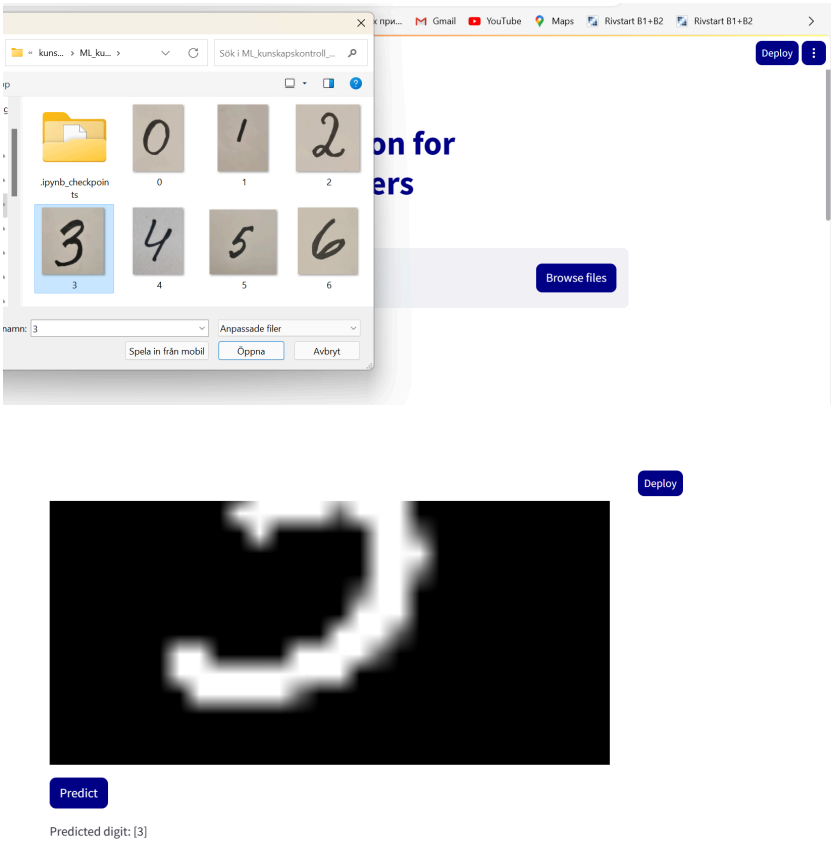
Figur 5. Förutsägelser som gjordes av Random Forest och K-nearest neighbors modellerna för fotografier som jag gjort och bearbetat själv

Man kan se i Figur 5 att K-nearest neighbors modellen gör korrekta förutsägelser, till skillnad från Random Forest

4.3. Streamlit application

För att skapa en Streamlit applikationen tränade jag modellen K-nearest neighbors separat på 50 000 observationer från MNIST-datan. Därefter sparade jag den här modellen med funktion joblib.dump.

Figur 6. Bilden som togs av Streanlit applikationen på en webbkamera



Figur 7. Bilden som som laddades upp av Streanlit applikationen

Med hjälp av Streamlit applikationen förutsägs siffrorna i bilderna ganska snabbt och exakt.

5. Slutsatser

Som resultatet av att ha genomfört den praktiska delen av kunskapskontroll kan jag göra följande slutsatser :

1. Alla tre modellerna visade ganska höga resultat när de tränade och testade med MNIST-data.
2. Den högsta noggrannheten under testen MNIST-data visade Random Forest model- 95,6%
3. Random Forest-modellen fungerade absolut inte för fotografier som jag tog och bearbetade själv.
4. För fotografier som jag tog och bearbetade själv använde jag K-nearest neighbors (KNN)-modellen.
5. Även om KNN-modellen inte visade samma precisionen som under testningen, gjorde den fortfarande korrekta förutsägelser med ett visst urval av fotografier.
6. Den svåraste delen av arbetet korrekt var urval och bearbetning av fotografier.
7. Streamlit applikationen förutsäger siffrorna i bilderna ganska snabbt och exakt.

6. Teoretiska frågor

1. Kalle delar upp sin data i ”Träning”, ”Validering” och ”Test”, vad används respektive del för?

Kalle ska använda “Träning” data att skapa flera olika modeller, ”Validering” data att välja den bästa modellen och “Test” data för att testa den bästa modellen.

2. Julia delar upp sin data i träning och test. På träningsdatan tränar hon tre modeller; ”Linjär Regression”, ”Lasso regression” och en ”Random Forest modell”. Hur skall hon välja vilken av de tre modellerna hon skall fortsätta använda när hon inte skapat ett explicit ”valideringsdataset”?

Julia kan använda “Cross Validation”, jämföra RMSE mellan de tre modellerna och välja den modellen med lägsta genomsnittliga RMSE-värdet.

3. Vad är ”regressionsproblem”? Kan du ge några exempel på modeller som används och potentiella tillämpningsområden?

”Regressionsproblem” betyder att baseras på en eller flera inputvariabler (X1, X2…Xn) kan vi prognasera en kontinuerlig variabel (y) som output.

Till exempel:

- Linjär regression användas att prognosera linjer samband mellan variabler. Potentiella tillämpningsområden kan vara en undersökning som visar sambandet mellan BMI och blodsocker

- Lasso regression som innebär regularisering för koefficienter som ger möjlighet att välja de mest betydelsefulla. Lasso regression kan användas för ekonomisk prognos som beror på många faktorer men Lasso regression ger möjlighet att välja de viktigaste.

- Support Vector Machines (SVM) används i regression att hitta en linje eller plan som separerar datapunkterna så bra som möjligt. SVM kan användas när man vill prognosera bostadspriser i olika områden.

4. Hur kan du tolka RMSE och vad används det till:

$$RMSE = \sqrt{\frac{1}{n}\sum_{1}^{n}(y_i - \hat{y}_i)^2}$$

Root Mean Squared Error (RMSE) används att göra en uppskattning av genomsnittlig avvikelse mellan de faktiska värdena från datan och de förutsagda värdena från modellen.

yi - de faktiska värdena från datan

$\hat{y}_i$ - de förutsagda värdena från modellen

n - antal beräkningar av en modell

5. Vad är ”klassificieringsproblem”? Kan du ge några exempel på modeller som används och potentiella tillämpningsområden? Vad är en ”Confusion Matrix”?

”Klassificieringsproblem” betyder att baseras på egenskaper eller attribut kan en modell skilja olika klasser åt och klassificera nya observationer.
Exempel på modeller:

- Logistisk regression heter "regression" men används faktiskt för klassificering. Logistisk regression visar sannolikheten för att en observation ingår i en viss klass. Potentiella tillämpningsområden för logistisk regression kan vara bedömning av risk för händelser.

- Support Vector Machines (SVM) används för att hitta en linje som separerar olika klasser. Potentiella tillämpningsområden för SVM kan vara medicinsk bildanalys eller bedömning av kreditrisk.

- Beslutsträd och beslutsgränsskikt delar upp datan i olika grenar baserat på olika egenskaper. Potentiella tillämpningsområden kan vara spamfiltrering

“Confusion Matrix” är en tabell som visar antalet korrekta och felaktiga svar och alltså utvärderar en klassificeringsmodell. Alla svaren som ger en modell delas:
- True Positive (TP) - antal korrekta positiva svar;
- False Positive (FP)- antal felaktiga positiva svar;
- True Negative (TN)- antal korrekta negativa svar;
- False Negative (FN)- antal felaktiga negativa svar.

6. Vad är K-means modellen för något? Ge ett exempel på vad det kan tillämpas på.

K-means är en algoritm som används för att gruppera datapunkter i k olika grupper (kluster) baserat på deras egenskaper eller attribut. Principen är att vi måste ange antalet kluster k som algoritmen ska hitta, efter det väljs en central punkt och efter det bildas en grupp av observationer runt den centrala punkten. Vi kan dela upp företagets kunder beroende på deras egenskaper till exempel inkomst.

7. Förklara (gärna med ett exempel): Ordinal encoding, one-hot encoding, dummy variable encoding. Se mappen ”l8” på GitHub om du behöver repetition.

Vid ordinal kodning tilldelas varje unik kategori en unik siffra eller en ordning.

För encoding skapar vi olika variabler till exempel:
Villa 1
radhus 2
lägenhet 3

1
2
3
2
2

Tabell 2. Encoding

För one-hot encoding skapar vi en ny binär variabel för varje unik kategori. Varje variabel är 1 om observationen tillhör kategorin och 0 annars till exempel:

Villa	radhus	lägenhet
1	0	0
0	1	0
0	0	1
0	1	0
0	1	0

Tabell 3. One-hot encoding

Dummy variable encoding löser problemet som vi har med one-hot encoding. Till exempel, om vi vet att [1, 0, 0] representerar "Villa" och [0, 1, 0] representerar "Radhus" behöver vi inte en annan binär

variabel för att representera "lägenhet", istället kan vi använda 0-värden för både "Villa" och "Radhus" enbart, t.ex. [0, 0].

Villa	Radhus
1	0
0	1
0	0
0	1
0	1

Tabell 4. Dummy variable encoding

8. Göran påstår att datan antingen är ”ordinal” eller ”nominal”. Julia säger att detta måste tolkas. Hon ger ett exempel med att färger såsom {röd, grön, blå} generellt sett inte har någon inbördes ordning (nominal) men om du har en röd skjorta så är du vackrast på festen (ordinal) – vem har rätt?

De båda har rätt. Datan verkligen är ”ordinal” (kategorisk data där det finns en inbördes ordning eller rangordning mellan kategorierna) eller ”nominal” (representerar kategorier eller grupper utan en inbördes ordning) dock i den här exemplen som ger Julia “röd” har en ordning kategori “vackrast”. Vi kan ställa skjortas färger i ordin till exempel röd är “vackrast” , näst vackraste gröna och minst vacker blå.

9. Kolla följande video om Streamlit: https://www.youtube.com/watch?v=ggDaRzPP7A&list=PLgzaMbMPEHEx9Als3F3sKKXexWnyEKH45&index=12 Och besvara följande fråga: - Vad är Streamlit för något och vad kan det användas till?

Streamlit är en källa med en kod som har öppen struktur för att skapa dataapplikationer i Python för maskininlärnings- och datavetenskapsteam. (Prgomet A., YouTube)

7. Självutvärdering

Utmaningar du haft under arbetet samt hur du hanterat dem.

Det största problemet var förbearbetningen av de fotografier som jag själv tog. Jag kan inte säga att problemet är helt löst. Random Forest modellen visade högsta resultaten på MNIST-data, men jag kunde inte tillämpa den på mina fotografier. Jag löste problemet helt genom att använda K-nearest neighbors (KNN)-modellen.

Vilket betyg du anser att du skall ha och varför.

G eller VG. Jag försökte uppfylla kriterierna för VG, men jag kan inte säga att jag själv är 100% nöjd med resultatet

8. Källförteckning:

1. Azure. Vad är maskininlärning? (Hämtad 13.03.2024) https://azure.microsoft.com/sv-se/resources/cloud-computing-dictionary/what-is-machine-learning-platform

2. Géron A. Hands-on Machine Learning with Scikit-Learn, Keras & TensorFlow beskrivs huvud. Concepts, Tools, and Techniques to Build Intelligent Systems . 2019 Kiwisoft S.A.S.

3. IBM. What is the k-nearest neighbors (KNN) algorithm?. (Hämtad 13.03.2024) https://www.ibm.com/topics/knn

4. Prgomet A. Introduction to Streamlit. YouTube. (Hämtad 13.03.2024) https://www.youtube.com/watch?v=ggDaRzPP7A&list=PLgzaMbMPEHEx9Als3F3sKKXexWnyEKH45&index=12