Machine Learning

Kunskapskontroll 2

rapport

Kamila Nigmatullina

EC – UTBILDNING, DS 2023

Uppdragsgivare: Antonio Prgomet

24 mars 2024

Innehåll

1. Inledning

I den här rapporten beskriver vi ett arbete med att skapa en maskinlärningsmodell som ska kunna prediktera MNIST-datasetet och testa om den kan även prediktera utomstående bilder.

Syftet med rapporten är att besvara följande frågeställningar:

1) Kan man skapa en maskinlärningsmodell som kan prediktera på MNIST dataset med minst 80%?
2) Kan man använda modellen successivt på utomstående bilder?

För att besvara andra frågan, skapade jag en app på Streamlit platformen.

2. Teori

LinearSVC

SVC - support vector machine utför algoritmer för både regressionsproblem och klassificering. Vi använder Linear hyperparameter i vårt arbete.

K-nearest neighbor (KNN)

KNN är en modell för både regressionsproblem och klassificering som är bra för att använda vid mönster-sökning.

MNIST

Modified National Institute of Standards and Technology (MNIST) är ett dataset som består av handskrivna siffror och används för att lära sig mönster-identifieringsmetoder.

För att fördjupa uppfattning av denna rapport, besvarar vi även några teoretiska frågor:

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

Träning: man fittar och tränar modellen. Validering: man justerar modellen på den. Test: man utvärderar modellens prestanda.

2. *Julia delar upp sin data i träning och test. På träningsdatan så 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”?*

Hon ska utvärdera modellerna på träningsdatan, men absolut inte röra testdatan eftersom resultat blir då inte objektiv.

3. *Vad är ”regressionsproblem? Kan du ge några exempel på modeller som används och potentiella tillämpningsområden?*
   Regression använs för att hitta förhållande mellan olika variabler. Exempel: Linjär regression, Lasso regression. Tillämpningsområden – medicin (till exempel, prediktera diabetes), prediktera priser beroende på olika faktorer.
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 Square Error – används för att se hur bra predikterar modellen. Ju högre värdet desto större är skillnaden mellan det predikterade värdet och det observerade värdet.

5. *Vad är ”klassificieringsproblem? Kan du ge några exempel på modeller som används och potentiella tillämpningsområden? Vad är en ”Confusion Matrix”?*
   Vid klassificeringsproblem predikteras datan enligt kategorier. Exempel av modeller: Logistisk regression, Random forest. Tillämpningsområde kan vara, till exempel, spam filter (spam / ej spam).

6. *Vad är K-means modellen för något? Ge ett exempel på vad det kan tillämpas på.*
   K-means är en algoritm som används vid klusteranalys. Principen är att gruppera datapunkter i olika kluster baserat på likheter mellan dem. Man kan använda den för att segmentera målgrupper vid marknadsanalays.

7. *Förklara (gärna med ett exempel): Ordinal encoding, one-hot encoding, dummy variable encoding.*
   Ordinal encoding – används för att omvandla kategoridka variabler till numeriska (till exempel, Green, Blue, Red – [0, 1, 2]). One-hot encoding – representerar varje värde med en binär vektor (till exempel, Green [1, 0, 0], Red [0, 1, 0]). Dummy encoding är en variant av one-hot encoding som man använder när antalet av kategorier är mer än två. (till exempel, Green, Blue, Red – dummy skulle skapa två binära variabler Green och Blue. Om både skulle vara 0, betyder der att färgen är Röd).

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?*
   Både och har rätt. Ordningen kan man sätta manuelt och detta blir då subjektivt eftersom inte alla tycker att man blir vackrast på festen i röd. Eller så kan der sorteras automatiskt i alphabetisk ordning, till exempel, i detta fall.

9. *Kolla följande video om Streamlit:*
   https://www.youtube.com/watch?v=ggDa□RzPP7A&list=PLgzaMbMPEHEx9Als3F3sKKXexWnyEKH45&index=12
   Och besvara följande fråga:
   - Vad är Streamlit för något och vad kan det användas till?
   Streamlit är en platform där man kan skapa olika app. Den är baserad på Pyton. Den innehåller inbyggda moduler så man slipper skapa appen från ”0”. Det är en bra lösning för att skapa prototyper utan att spändera mycket tid och pengar på det.

3. Metod, resultat och slutsatser

Dataset MNIST var laddat.

Träning och test

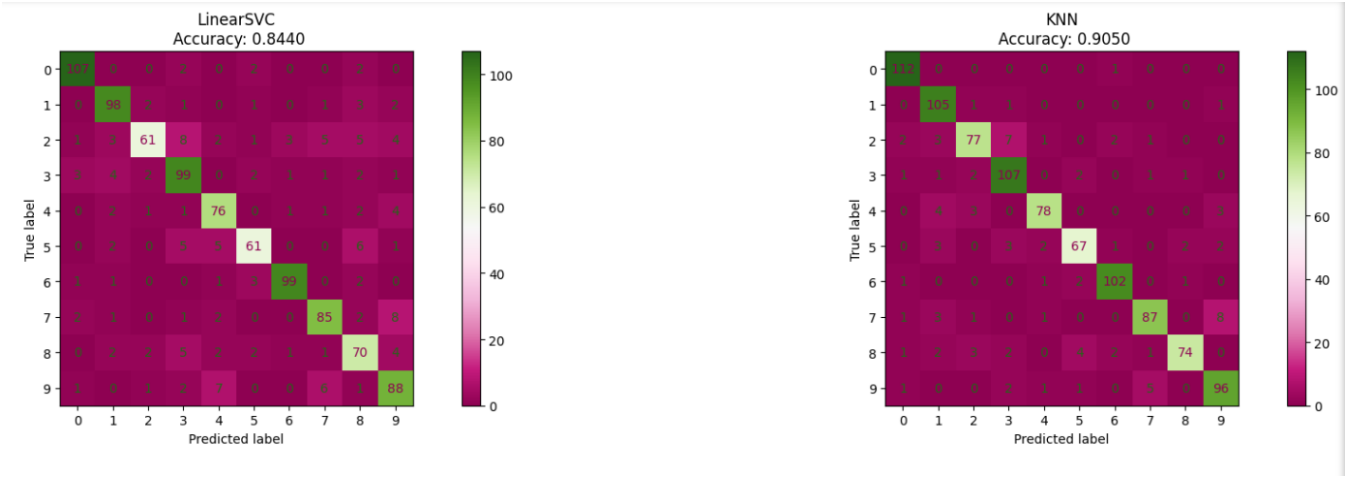
Träningsdata skalades ner för att träningsprocessen går fortare.

Vi tränade två modeller: LinearSVG och KNN.

```
LinearSVC
LinearSVC(random_state=42)
```

```
KNeighborsClassifier
KNeighborsClassifier()
```

Vid utvärdering av modeller, visade det sig att accuracy vid KNN algoritmen är högre.



Vi tog ett besked att köra vidare med KNN modellen och finjustera den med hjälp av GridSearch.

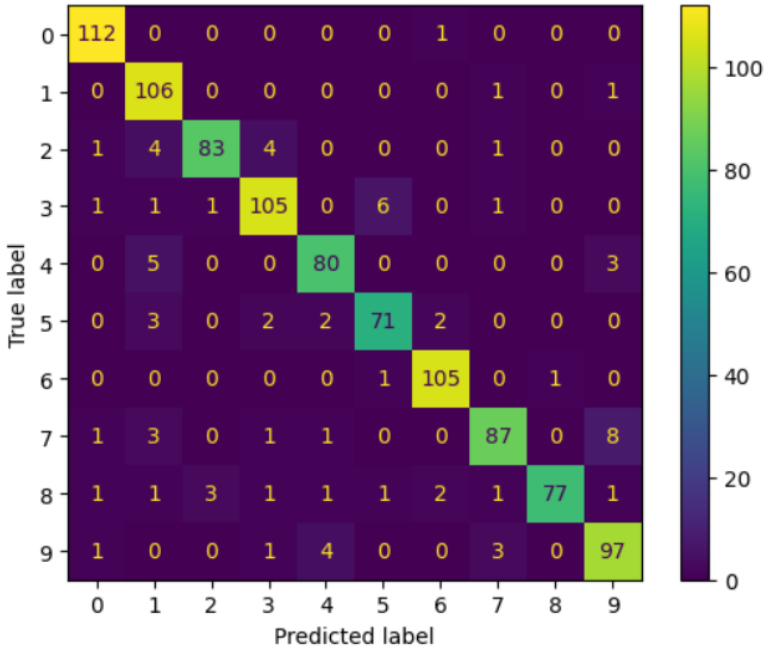
Vi satt några hyper-parametrar att välja emellan:

Best parameters: {'leaf_size': (20, 40, 1), 'metric': ('minkowski', 'chebyshev'), 'n_neighbors': (1, 10, 1), 'p': (1, 2), 'weights': ('uniform', 'distance')}

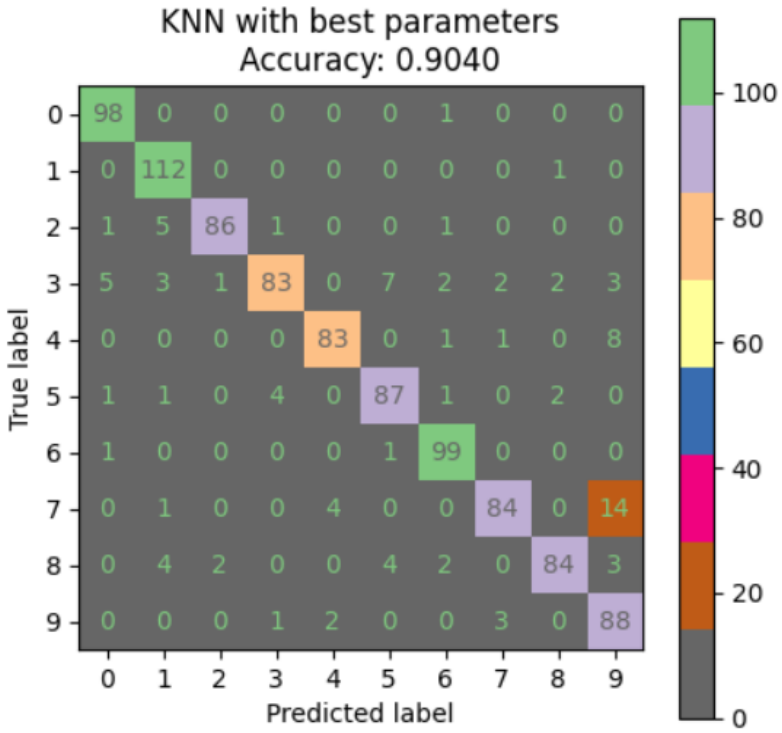
Bästa hyperparametrar som vi har utvärderat med hjälp av GridSearch:

```
{'leaf_size': 20, 'metric': 'minkowski', 'n_neighbors': 1, 'p': 1, 'weights': 'uniform'}
```

Accuracy vid prediktionen på valideringsdata visade sig vara 92%. Vi visaliserade också förm ågan med hjälp av Confusion matrix.



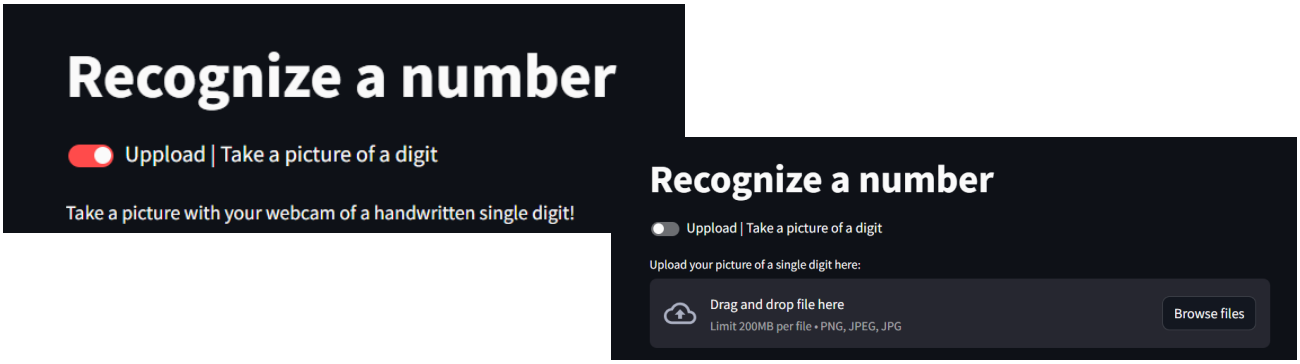
Vi gjorde prediction på testdatan med vår finjusterade KNN-modellen. Accuracy blev då 90% som visar att vi har uppnått vårt första mål.



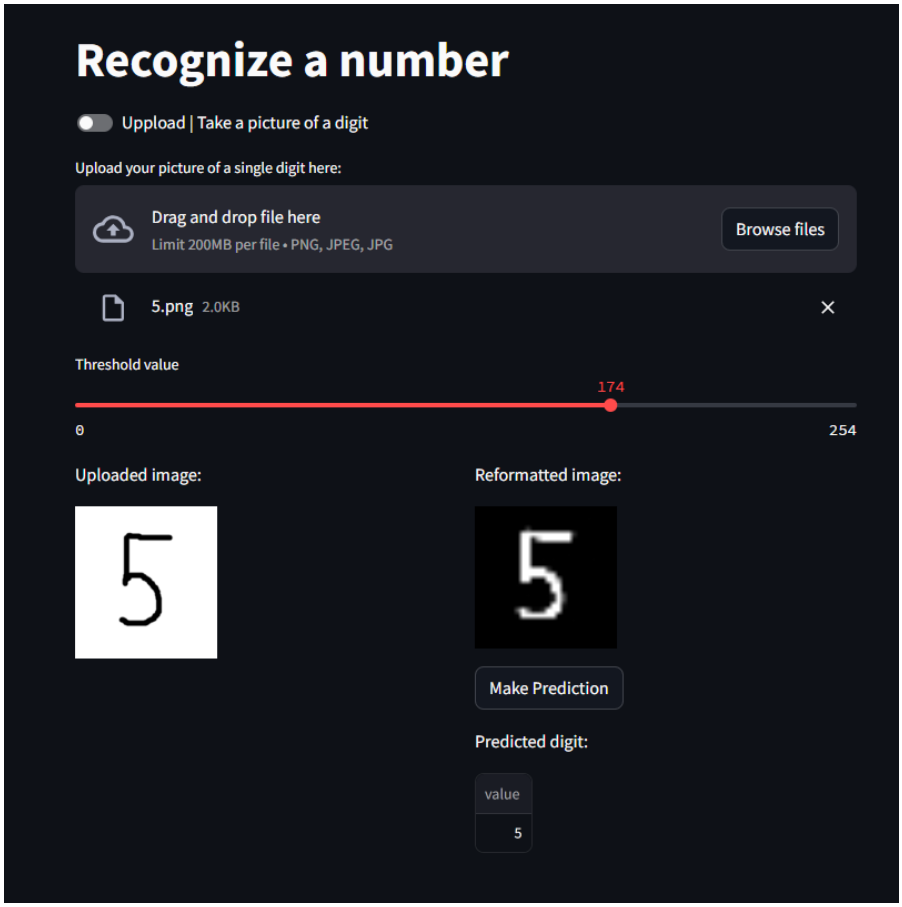
3.1. Streamlit

Vi har skapat en enkel app där man identifiera handskrivna siffror.

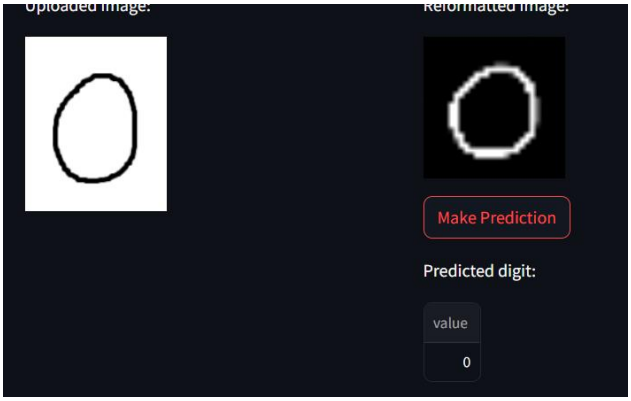
Man kan både ta kort med kameran och ladda upp bilder.

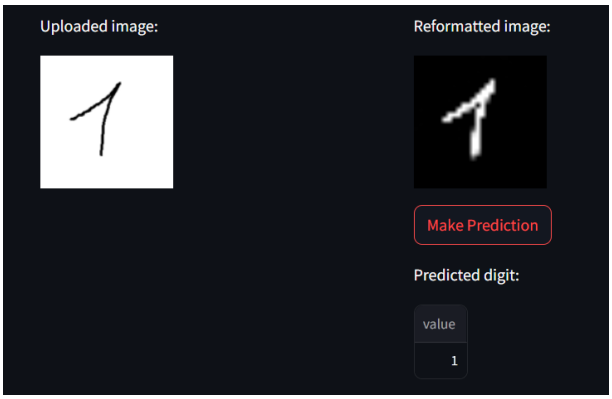


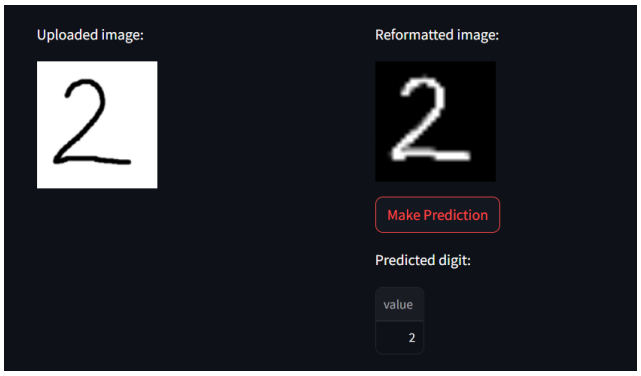
3.1.1. Interface:

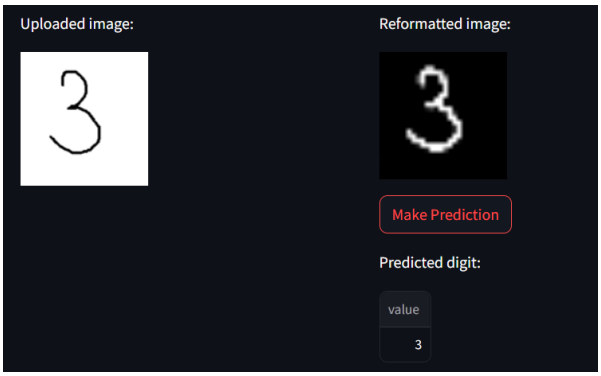


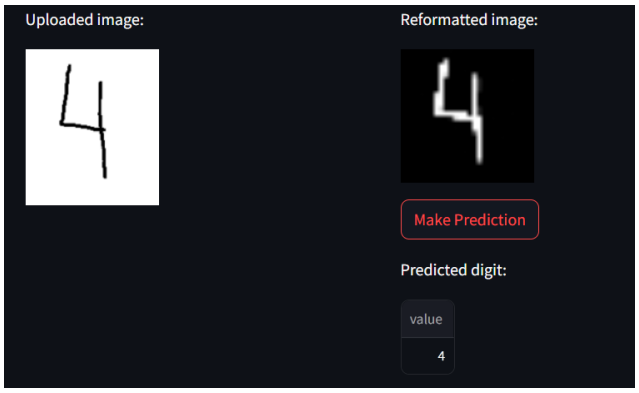
Siffrorna som har successivt predikerats:

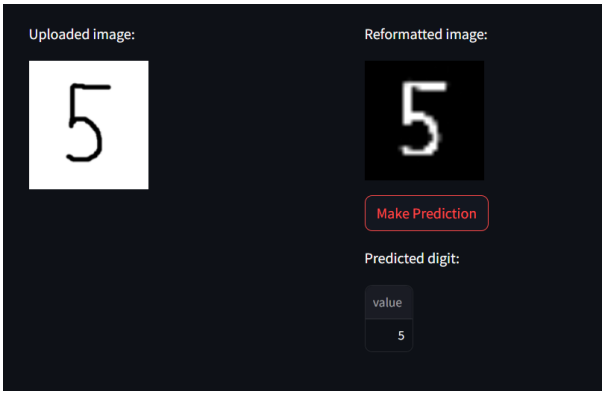


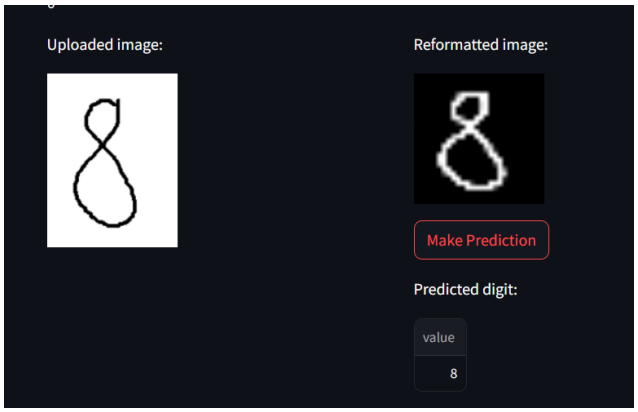


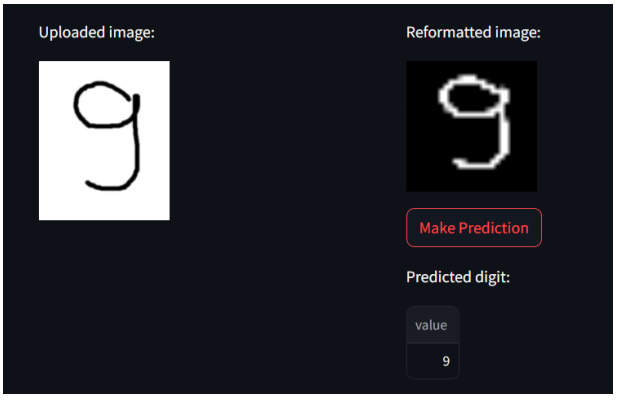




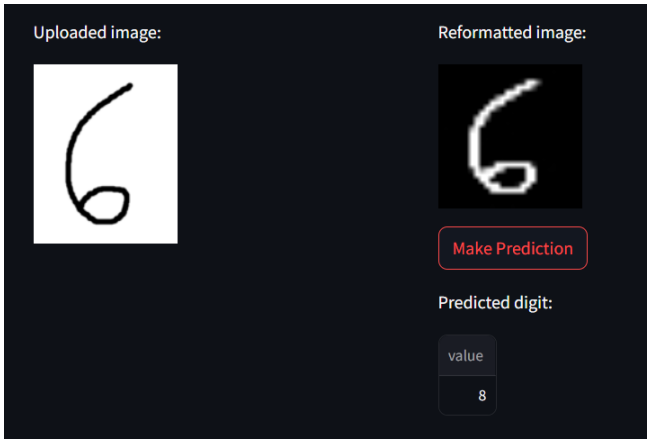


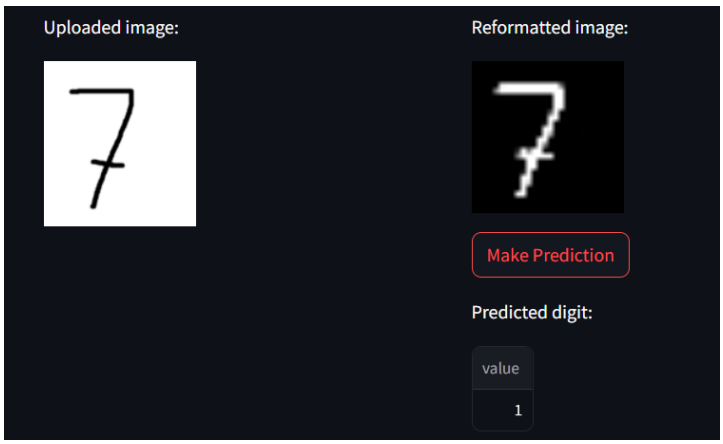






Siffrorna som predikerades fel:





Man ser att modellens prestanda grovt blev 80%. Det betyder att vi har nått vårt andra mål, men detta projekt måste utvecklas vidare.

4. Diskussion

Vi har testat vald modellen både på dataset och genom appen med egna bilder. Grovt accuracy vid testkörning på appen blev 80% som är mindre i jämförelse med testkörningen på MNIST-datasetet.

Vi upptäckte att detta kan bero inte på modellen utan på själva bilder. Skriver man med tjockare penna – predikteras det bättre. Bilder som vi inte kunde prediktera var 6 och 7. Detta måste undersökas och det kan ha med preprocessing av bilder att göra.

5. Slutsatser

Generellt har vi nått våra mål:

1) Skapade en fungerande modell med accuracy högre än 80% (90%).
2) Vi upptäckte att modellen fungerar även med utomstående bilder. Prestandan är dock sämre i detta fall. Praktiskt predikterade bara modellen 8 av 10 siffror.

Vi behöver undersöka vidare frågan angående preprocessing av bilder för att förbättra vår projekt och höja appens prestandan.

6. Källförteckning

1) https://towardsdatascience.com/everything-about-svm-classification-above-and-beyond-cc665bfd993e
2) https://www.geeksforgeeks.org/k-nearest-neighbours/
3) https://paperswithcode.com/dataset/mnist
4) https://docs.opencv.org/