Kunskapskontroll

Machine learning

Mustafa almazerli

EC Utbildning

Machning learning

202403

Abstract

This report provides an overview of my exploration into the realm of machine learning, delving into fundamental concepts and techniques. Through practical applications and hands-on exercises, I navigated the landscape of algorithms, focusing on their implementation and optimization. The journey encompassed key topics such as classification, regression, and clustering. The report highlights the significant results achieved in terms of model accuracy and predictive capabilities. Overall, this study serves as a foundational step in understanding the intricate dynamics of machine learning, paving the way for future exploration and application in real-world scenarios.

Förkortningar och Begrepp

Maskininlärning (ML)
K-nearest neighbors (KNN)

Super Vector Machine (SVM)

Innehållsförteckning

Abstract 2
Förkortningar och Begrepp Maskininlärning (ML) 3
1 Inledning 1
1.1 Underrubrik – Exempel 1
2 Teori 2
2.1 Random forest 2
2.2 K-nearest neighbors 2
2.3 Neural Network-modell 2
2.4 Super Vector Machine 2
3 Metod 3
3.1 Datamodellering 3
3.2 Testkörning 3
3.3 Utvärdering och välja modell 3
3.4 Slutlig träning 3
3.5 Streamlit och OpenCV 3
4 Resultat och Diskussion 4
4.1 TEST AV MODELLER 4
4.2 Test av egna bildar 4
4.3 Streamlit 4
5 Slutsatser 6
5.1 fråga 1: 6
5.2 fråga 2 6
6 Teoretiska frågor 7
7 Självutvärdering 9
Appendix A 10
Källförteckning 11

1 Inledning

Maskininlärning (ML) har blivit en central del av vår digitala värld och omfattar en mängd olika tekniker som möjliggör datorer att lära sig från data och utföra uppgifter utan att vara explicit programmerade. Denna revolutionerande disciplin har visat sig vara särskilt kraftfull när det gäller att lösa komplexa problem och göra prediktioner baserat på data.

Syftet med denna rapport är att skapa en maskinlärningsmodell för att prediktera MNIST datasetet och prediktera bildar från webkamera eller mobilkamera.

, för att uppfylla syftet så kommer följande frågeställningar att besvaras:

1. Skapar en modell som kan prediktera MNSIT mer än 90%
2. Kan modellen prediktera bildar från webkamera eller mobilkamera.

1.1 Underrubrik – Exempel

För andra frågan kommer skapa en applikation genom Streamlit och openCV för att prediktera handskrivna siffror.

2 Teori

2.1 Random forest

Random Forest-modellen använder en ensemble av beslutsträd för att förbättra prediktionskraften och minska överanpassning. Genom att kombinera resultaten från flera träd ger den robusta och precisa förutsägelser för olika maskininlärningsproblem. (Breiman, 2001).

2.2 K-nearest neighbors

K-nearest neighbors (KNN) är en icke-parametrisk, övervakad inlärningsalgoritm som använder närhetsinformation för att göra klassificeringar eller förutsägelser om gruppering av en individuell datapunkt.

2.3 Neural Network-modell

Neurala nätverk är kraftfulla maskininlärningsmodeller som kan extrahera komplexa mönster från data, särskilt lämpliga för bildigenkänning och MNIST-datasetet.

2.4 Super Vector Machine

Super Vector Machine (SVM) är en övervakad inlärningsalgoritm som används för klassificering och regression. Den används effektivt för att skapa en optimal separerande hyperplan mellan olika klasser av datapunkter, vilket gör den särskilt användbar för att identifiera handskrivna siffror i MNIST-datasetet (Cortes et al., 1995).

3 Metod

För att uppnå syften med att skapa en modell som kan prediktera MNIST-datastet med en noggrannhet på över 90% och möjliggöra prediktera av bildar från webbkamera eller mobilkamera, kommer följande metod används:

3.1 Datamodellering

Datainsamling och förberedelse: MNIST -datasetet kommer att användas för träning och utvärdering av modellerna. För bildprediktion från webbkamera eller mobilkamera kommer OpenCV att användas för att fånga förbereda bilder för analys.

3.2 Testkörning

I den inledande fasen av testkörningen användas samtliga maskinlärningsaalgoritmer med deras standardparametrar för att får en övergripande uppfattning om vilka modeller som presterade bäst på MNIST-datasetet. Träningsdatamängdarna skalades ner för att underlätta bearbetningen och spara tid, till början användas hela datasetet, men på grund av långa bearbetningstiden använda en de av datamängden. Därefter gjordes förutsägelser med flera maskininlärningsalgoritmer, inklusive Random forest, K-NN, Neurala nätverk och Super Victor Machines på testdata för att utvärdera och identifiera den bästa modell som uppfylla kraven på över 90% noggrannhet för MNIST -prediktioner.

3.3 Utvärdering och välja modell

Modellerna kommer att utvärderas genom att modellerna tränats och testats på valideringsdata och testdata, därefter har beslutet att använda Random Forest som presterade bättre an andra modeller. För att ytterligare förbättra den valda modellens prestanda genomfördes en GridSearch cross-validation för hyperparametrarna "C" och "Gamma". Parametern för GridSearch var "scoring = accuracy.

3.4 Slutlig träning

De bästa hyperparametrarna som valts användes för att träna en ny Random Forest-modell på hela datasetet bestående av 70 000 bilder. Efter att modellen tränats färdigt sparades den för användning i Streamlit-applikationen.

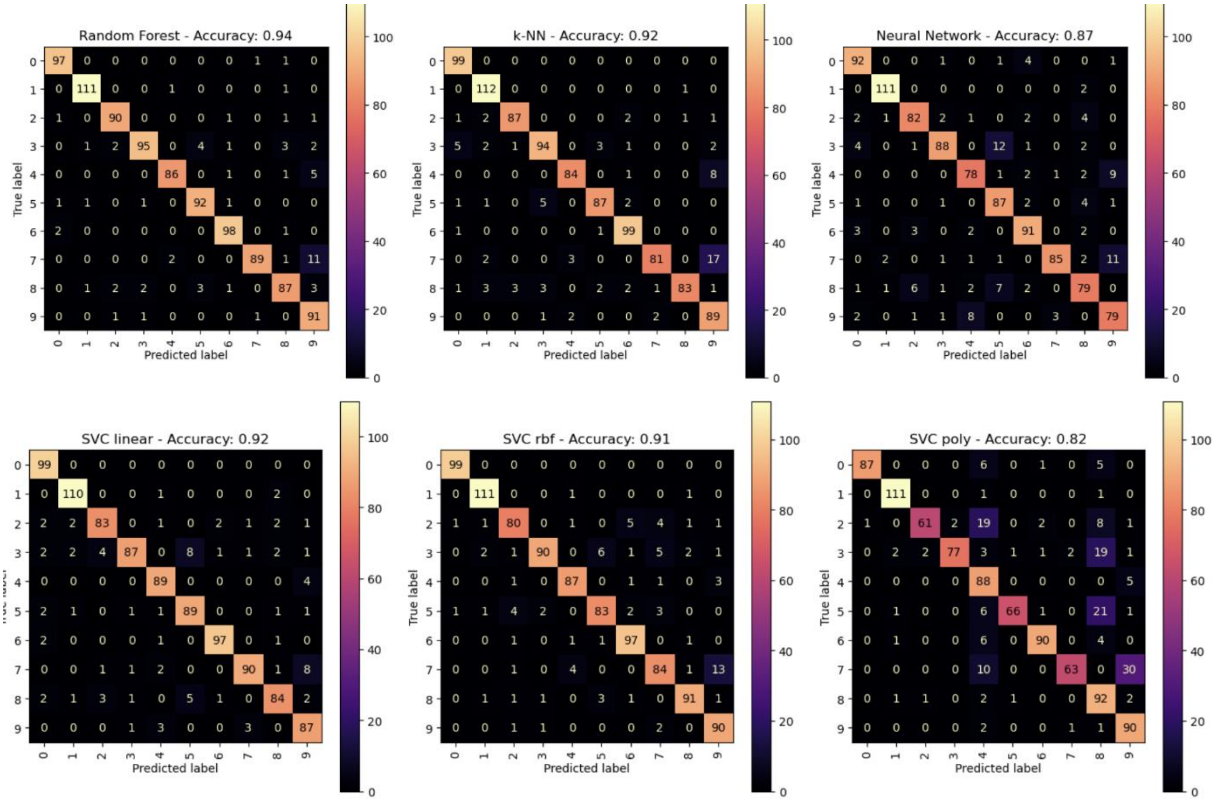
3.5 Streamlit och OpenCV

Innan projektet påbörjades fanns ingen tidigare erfarenhet av vare sig Streamlit eller OpenCV. Jag ägnade några dagar åt att lära mig grunderna och hitta lämpliga resurser för att genomföra projektet. Applikationer kommer att utvecklas med hjälp av Streamlit och OpenCV, och för att säkerställa dess funktionalitet utförde jag tester med olika metoder. Koden skrevs om flera gånger då resultaten var inte rätt till början.

4 Resultat och Diskussion

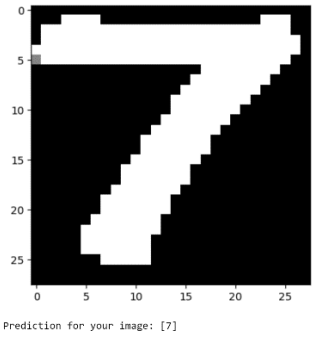
4.1 TEST AV MODELLER

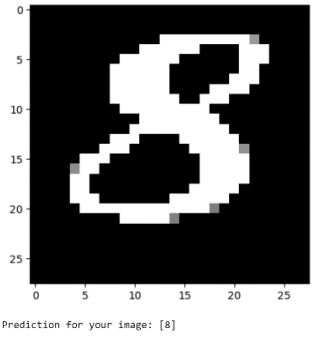
Efter att ha analyserat förutsägelsematriserna för varje modell framgår det att Random Forest-modellen har den minsta mängden felaktiga förutsägelser. Detta innebär att den modellen har den bästa prestandan när det gäller att korrekt klassificera bilder av handskrivna siffror.

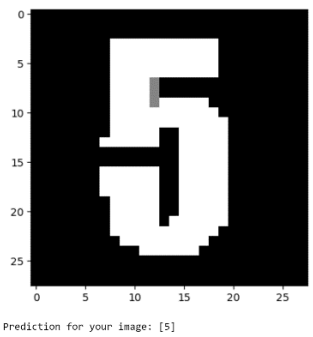


4.2 Test av egna bildar

Efter att ha tränat den valda Random Forest-modellen på hela datasetet och tillämpat de optimerade hyperparametrarna, observerades en förbättring av dess prestanda. Genom att använda den slutgiltiga modellen för förutsägelser på testdata enligt den beskrivna metoden, kunde jag validera dess höga noggrannhet och effektivitet för att prediktera handskrivna siffror. Här kommer några exempel dessa predation var rätt efter många försöka, modellen hade svårt min vissa färger och storlek därför fick test många olika bildar för att för rätt prediktion.







4.3 Streamlit

Streamlit-applikationsprocessen börjar med att testa många olika kodsnuttar och försöka skriva ett fungerande skript. Under tiden jag skriver koden fick jag hjälp av en klasskamrat att testa skriptet. Problemen var inte direkt i koden, utan jag behövde spara min tränade modell med jobb-funktionen

på rätt sätt, även modellen fick träna om många gånger för jag får alltid fel prediktion, har testat med att träna andra modeller på hela data för att för bättre svar men det är ju inte användbar på grund av tidsmässig. Efter många försöka med olika siffra fick till slut en rätt svar men att får rätt igen var stor utmaning. Det kan beror på bilderna, fäger eller min kamera men för att får rätt svar de hade modellen behvöt träna på mer data eller hitta bättre kamera och rätt miljö för bildar

5 Slutsatser

5.1 att skapa en modell som kan predikterar MNIST data med mer än 90%:

Ja genom användning av olika maskininlärningsalgoritmer och noggrann utvärdering av deras prestanda kunde en Random forest – modell identifieras som uppnådde en noggrannhet på över 90% vid prediktion av MNIST -datasetet. Detta uppfyller målet men 0,94% noggrannhet vilken var bättre än målet.

5.2 kan modellen prediktera bildar från webbkamera eller mobilkamera

Ja, målet är uppfylld även om det var en utmanade process att anpassa modellen för att hantera olika typer av bildar och miljöer. Trots visa svårigheter med olika koder, bildar, färger, storlekar och kvalitet på bildarna kunde modellen ge tillförlitliga prediktioner när den väl kunde läser bildarna.

Sammanfattningsvis visare slutsatserna att båda målet att uppnå noggrannhet på över 90% och utveckla en fungerande applikation för att predikterar bildar har uppnåtts. Detta är ett positiv resultat som visar potentialen och användbarheten hos maskininlärning och bildigenkänningstekniker i praktiska tillämpningar.

6 Teoretiska frågor

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

Kalle genomför denna process för att hitta optimala modellen för sina data. Första steget, Träning, skapar Kalle två eller flera modeller och tränar dem. Andra steget ät Validering, där Kalle utvärderar och jämför kapaciteten hos de tränade modellerna med hjälp av en sparat valideringsuppsättning. Slutligen, i det tredje steget, Test, testar Kalle den valde modellen på data för att säkerställa att den fungerar bra, målet med sista steg är att skapa en stark och pålitlig modell.

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 ”validerings datasats”?**

Julia kan använda sig av’ Cross-Validation’ där använder hon metoden K-faldig för att träna data K gånger för att utvärderas resultatet på de tre modellerna som tränings, och välja den bästa.

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

Regressionsproblem uppstår när vi baserat på en eller flera inputvariabler (x1, x2, ..., xn) försöker prognostisera en kontinuerlig variabel y som output. Ett exempel på en regressionsmodell är Lasso Regression, som använder regularization för att hantera koefficienter och möjliggör urval av de mest betydelsefulla variablerna. Potentiella tillämpningsområden för regressionsproblem inkluderar ekonomi, medicin och marknadsföring

4. **Hur kan du tolka RMSE och vad används det till: ** $MSE = \sum(yi - \hat{y}i)2\ i$

RMSE, eller Root Mean Squared Eror, är ett mått inom regression som beräknar det genomsnittliga felet mellan förutsagda och faktiska värden, för att talka RMSE behöver man förstår två huvudpunkter, först ju lägre RMSE-värdet är, desto bättre passar modellen data. Det betyder att avvikelsen mellan det vi förutspår och det faktiska värdet är mindre. För det andra ger RMSE en känsla av hur mycket de förutsagda värdena avviker från de faktiska värdena, så målet är att ha så lågt RMSE-värde som mögligt.

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

Klassificieringsproblem är att kategorisera data i olika klasser eller kategorier till vilka kategori eller klass en given observation tillhör.

Exempel: Logistisk regression och support Victor Machines(svm).

potentiella tillämpningsområden: spamfiltrering: att klassificera e-post som antingen spam eller inte.

Confusion Matrix ”är en tabell som används för att utvärdera prestandan hos en klassificeringsmodell. Den visar antalet korrekta och felaktiga klassificeringar gjorda av modellen jämfört med de faktiska klasserna.

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

K-menas är en klusteringsalgoritm som grupperar datapunkter i k olika grupper baserat på deras egenskaper eller attribut. Vi måste ange antalet kluster, k, som algoritmen ska hitta. Algoritmen väljer centrala punkter och bildar grupper av observationer runt varje central punkt. Ett exempel på tillämpning är En e-handelsplattform använder K-means för att gruppera produkter baserat på kunders köpbeteende

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

Alla tre tekniker som används för att omvandla kategoriska data till numerisk form, men på olika sätt, om man ta till exempel värdet, så ordinal encoding kommer omvandla solig till 1 molnig 2 och regnigt till 3 det beror på att metoden går efter ordning. Medan one-hot encoding kommer skapar binära värde för varje kategori:

Soligt, molnigt och regnigt

0	0	1
0	1	0
1	0	0

Dummy coding är nästan like one-hot men det kommer använder mindre binära genom att utesluta en kategori, dummy kan använder data med ordning. I exemplet ovan använder one. Hot 3 olika värde medan här kommer blir

väder	molnigt	regnigt
Soligt	0	0
molnigt	1	0
regnigt	0	1

8. **Göran påstår att daten antingen är ”ordinal” eller ”nominal”. Julia säger att detta måste tolkas. Hon ger ett exempel med att färger såsom {grön, röd, 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?**

Båda har rätt men det beror på sammanhanget, generellt sett är att färger är nominala, men i vissa sammanhang kan tolkas som ordinal om en ordning ge i vissa tillfällen mer än andra färger som en röd skjorta på festen normalt kommer du vara mer uppmärksamhet med röd skjorta än vit eller svart och det beror på att du kanske var den enda som har röd skjorta

9. Vad är Streamlit för något och vad kan det användas till?

Streamlit är ett ramverk med öppne källkod för att skapa dataapplikationer i pyhton för maskininlärnigs och datavetenskpsteam**.**

7 Självutvärdering

1. Utmaningar du haft under arbetet samt hur du hanterat dem.

   Under arbetet stötte jag på många utmaningar en av de var att välja lämpliga hyperparametrar och att förstå valideringsprocessen för modellerna. För att hantera detta studerade jag dokumentationen noga, använde GridSearchCV för hyperparameteroptimering och sökte hjälp från kollegor och onlineforum för råd när jag behövde det. Denna strategi hjälpte mig att övervinna svårigheterna och förbättra mina färdigheter inom maskininlärning.

2. Vilket betyg du anser att du skall ha och varför.

Jag anser att jag förtjänar ett betyg som återspeglar mina ansträngningar och resultat. Genom att tillämpa noggranna forskningsmetoder och visa uthållighet och engagemang för att lösa problem har jag framgångsrikt producerat en rapport och en applikation som uppfyller de givna kraven.

3. Något du vill lyfta fram till Antonio?

Jag vill tacka Antonio stöd och den vägledning jag har fått under projektets gång, jag uppskattar verkligen din förmåga att förklara komplexa ämnen på ett begripligt sätt.

Appendix A

```
# 1. Random Forest-modell
random_forest_clf = RandomForestClassifier(n_estimators=100, random_state=42)
random_forest_clf.fit(X_train, y_train)

# Utvärdera på valideringsdata
rf_val_predictions = random_forest_clf.predict(X_val)
rf_val_accuracy = accuracy_score(y_val, rf_val_predictions)
print(f"Random Forest Validation Accuracy: {rf_val_accuracy:.2f}")
# Gör förutsägelser på testdatan
rf_test_predictions = random_forest_clf.predict(X_test)

# Utvärdera prestanda på testdatan
rf_test_accuracy = accuracy_score(y_test, rf_test_predictions)
print(f"Random Forest Test Accuracy: {rf_test_accuracy:.2f}")
```

```
C:\Users\musta\anaconda3\Lib\site-packages\sklearn\base.py:1151: DataConversionWarning
array was expected. Please change the shape of y to (n_samples,), for example using ra
  return fit_method(estimator, *args, **kwargs)
```

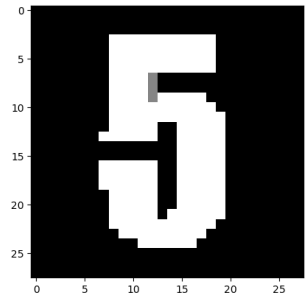
```
Random Forest Validation Accuracy: 0.94
Random Forest Test Accuracy: 0.94
```

```
Best parameters for RandomForestClassifier: {'max_depth': 20, 'n_estimators': 150}
Best cross-validation score: 0.93
```

```
# Gör förutsägelser på testdatamängden
predictions = random_forest_clf.predict(X_test)

accuracy = accuracy_score(y_test, predictions)
print(f"Model Accuracy on Test Data: {accuracy:.2f}")
```

```
Model Accuracy on Test Data: 0.97
```

```
lower_pixel = 100
upper_pixel = 130

image1 = Image.open(r"C:\Users\musta\OneDrive\Skrivbord\bildar\Skärmbild 2024-03-17 092425.png")

#  OpenCV-format
image_test = cv2.cvtColor(np.array(image1), cv2.COLOR_RGB2BGR)

# ändra storlek
gray_image = cv2.cvtColor(image_test, cv2.COLOR_BGR2GRAY)
resized_image = cv2.resize(gray_image, (28, 28))

for i in range(resized_image.shape[0]):
    for j in range(resized_image.shape[1]):
        if resized_image[i, j] <= lower_pixel:
            resized_image[i, j] = 0
        elif resized_image[i, j] > upper_pixel:
            resized_image[i, j] = 255

flattened_image = resized_image.flatten().reshape(1, -1)

# Visa bilden
plt.imshow(resized_image, cmap=mpl.cm.binary)
plt.show()

# Gör en förutsägelse med modellen
prediction = random_forest_clf.predict(flattened_image)

# Visa resultatet
print(f"Prediction for your image: {prediction}")
```




```
Prediction for your image: [5]
```

Källförteckning

(LeCun, 1998)Breiman, L. (2001). Random forests. Machine learning, 45(1), 5-32.

, C., & Vapnik, V. (1995). Support-vector networks. Machine learning, 20(3).

(johsnon, 2022the impact of artifical intelligence on business innovation)

Hastie, T., Tibshirani, R., & Friedman, J. (2009). The elements of statistical learning: data mining, inference, and prediction. Springer Science & Business Media.

Goodellow, 1, bengio & courville A. (2016). Deep learning. MIT press.

Cortes, C &

https://scikit-learn.org/stable/index.html

https://docs.streamlit.io/

https://juejung.github.io/jdocs/Comp/html/Slides_MachineLearning_1.html#inspecting-the-data