Machine Learning

Kunskapskontroll 2

ECUTBILDNING

Mohamad Almazrli

EC Utbildning

202403

Abstract

This project focuses on leveraging machine learning techniques to model the MNIST dataset, a collection of handwritten digits. The task involves evaluating at least two different models through a complete ML pipeline, starting from data loading to the evaluation of the best-performing model on test data. The process encompasses data acquisition using standard datasets like MNIST, exploration, preprocessing, model training, hyperparameter tuning, and model evaluation. Through this endeavor, the goal is to demonstrate the efficacy of machine learning in accurately predicting handwritten digits and showcase the importance of model selection and optimization in achieving optimal performance.

Innehåll

Intending

Denna rapport fokuserar på att utforska maskininlärning genom att utveckla en modell för att förutsäga MNIST-datasetet, vilket består av handskrivna siffror. Vi strävar efter att skapa en modell med en träffsäkerhet på minst 80% och utforska dess förmåga att generalisera till externa bilder från källor som webbkameror och sparade bilder. Genom att utveckla en applikation med Streamlit och OpenCV kommer vi att testa modellens generaliseringsförmåga. Syftet är att bidra till förståelsen inom maskininlärning och bildigenkänning samt utforska dess tillämpningsområden.

1. Är det möjligt att skapa en modell som kan förutsäga MNIST minst 80%?

2. Kan modellen förutsäga bilder från antingen dators webbkamera eller från sparade bilder?

I denna kunskapskontroll kommer därefter nio utvalda frågor att besvaras längst ner.

2 Teori

2.1 Modeller och dataset

RandomForestClassifier:

RandomForestClassifier är en ensemble-algoritm som består av flera beslutsträd. Varje träd i skogen utför en klassificering, och det mest populära klassresultatet väljs som det slutliga resultatet. För initial träning används standardinställningar.

KNeighborsClassifier:

KNeighborsClassifier är en enkel klassificeringsalgoritm som baseras på att hitta de närmaste datapunkterna i träningsdatamängden. Valet av antal grannar (k) kan påverka algoritmens prestanda och kan justeras under träning.

Logistic Regression:

Logistic Regression är en klassificeringsalgoritm som används för binär klassificering, vilket innebär att den förutsäger sannolikheten för att en given observation tillhör en viss klass. Den producerar resultat som ligger mellan 0 och 1 genom att använda en logistisk sigmoid-funktion. För initial träning används standardinställningar.

Dataset:

MNIST-datasetet är det valda datasetet för detta arbete. Det innehåller 70 000 bilder av handskrivna siffror i 28x28-pixelformat. Detta dataset har varit en benchmark för maskininlärningsalgoritmer i många år och används ofta för att utvärdera prestanda för olika modeller.

Metod:

För att svara på frågeställningarna och utveckla en modell för att prediktera MNIST-datasetet samt undersöka om modellen kan prediktera bilder från datorns webbkamera eller sparade bilder.

3.1Datainsamling och förberedelse:

För detta projekt kommer MNIST-datasetet att användas för träning och utvärdering av modellerna. MNIST-datasetet är en välkänd samling handskrivna siffror i bildformat och är vanligt förekommande inom maskininlärning för att testa och utveckla olika algoritmer för bildklassificering. För att möjliggöra bildprediktion från webbkamera kommer OpenCV att användas för att fånga och förbereda bilder för analys. OpenCV är ett kraftfullt bibliotek för bildbehandling och datorseende som ger möjlighet att interagera med kameror och bearbeta bilddata i realtid. Genom att använda OpenCV kan vi skapa en applikation som kan ta emot bilder från en webbkamera, förbereda dem för analys och sedan använda våra maskininlärningsmodeller för att förutsäga de handskrivna siffrorna i bilderna.

3.2 Testkörning

Efter att ha kört och utvärderat modellerna sparades de för framtida användning för att spara tid och resurser vid kommande tester eller justeringar. Detta möjliggjorde snabbare iterationer och experiment i projektet.

3.3 Vald modell

Efter en omfattande utvärdering visade både RandomForestClassifier och KNeighborsClassifier en imponerande prestanda med en högsta noggrannhet på 0,94, vilket var det bästa resultatet bland alla testade modeller. Denna höga noggrannhet och prestanda bekräftar deras förmåga att effektivt hantera och prediktera MNIST-datasetet, vilket är avgörande för att uppnå målen i projektet. Därför bekräftades valet av dessa modeller som de mest lämpliga för fortsatt användning och optimering.

3.4 Streamlit och OpenCV

Innan projektet påbörjades hade jag ingen tidigare erfarenhet av vare sig Streamlit eller OpenCV. Jag lärde mig grunderna på några dagar och utvecklade en applikation för att prediktera handskrivna siffror med hjälp av dessa verktyg. Genom tester och justeringar skapade jag ett användarvänligt gränssnitt för att ladda upp modeller och utföra prediktioner med antingen webbkamera eller egna bilder.

4 Resultat och Diskussion

4.1 Test av modeller

Alla SVS-modeller presterade bäst bland de primära testerna, men skillnaderna i prestanda var förhållandevis små, med bara en marginell skillnad på någon procentenhet

```
C:\Users\46736\anaconda3\Lib\site-packages\sklearn\utils\validation.py:1184: DataConversionWarning: A column-vector y was passe
d when a 1d array was expected. Please change the shape of y to (n_samples, ), for example using ravel().
  y = column_or_1d(y, warn=True)
```

```
Model SVC - Accuracy: 0.92
```

```
C:\Users\46736\anaconda3\Lib\site-packages\sklearn\utils\validation.py:1184: DataConversionWarning: A column-vector y was passe
d when a 1d array was expected. Please change the shape of y to (n_samples, ), for example using ravel().
  y = column_or_1d(y, warn=True)
```

```
Model SVC - Accuracy: 0.91
```

```
C:\Users\46736\anaconda3\Lib\site-packages\sklearn\utils\validation.py:1184: DataConversionWarning: A column-vector y was passe
d when a 1d array was expected. Please change the shape of y to (n_samples, ), for example using ravel().
  y = column_or_1d(y, warn=True)
```

```
Model SVC - Accuracy: 0.82
```

Baserat på analysen av förvirringsmatriserna för varje modell är det tydligt att både Random Forest-modellen och k-NN-modellerna uppvisar den minsta andelen felaktiga förutsägelser. Detta indikerar att både Random Forest och k-NN-modellerna presterar på en hög nivå när det gäller korrekt klassificering av bilder med handskrivna siffror.

```
k-NN Validation Accuracy: 0.94
```

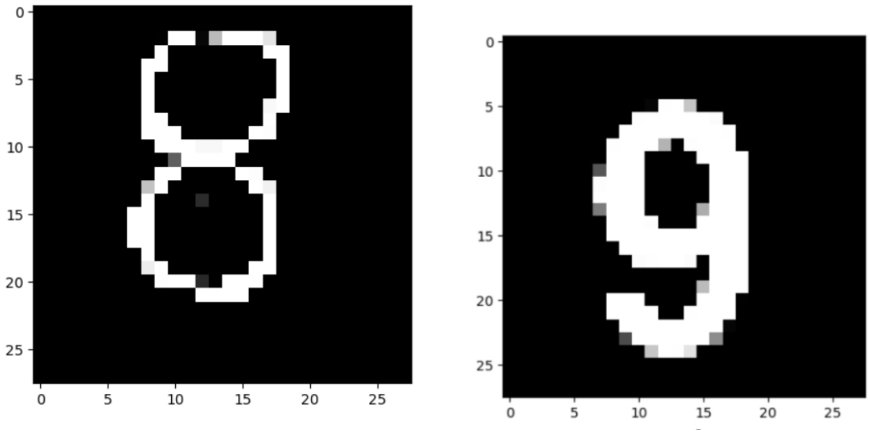
```
Random Forest Validation Accuracy: 0.94
```

4.2 Test av egna bildar

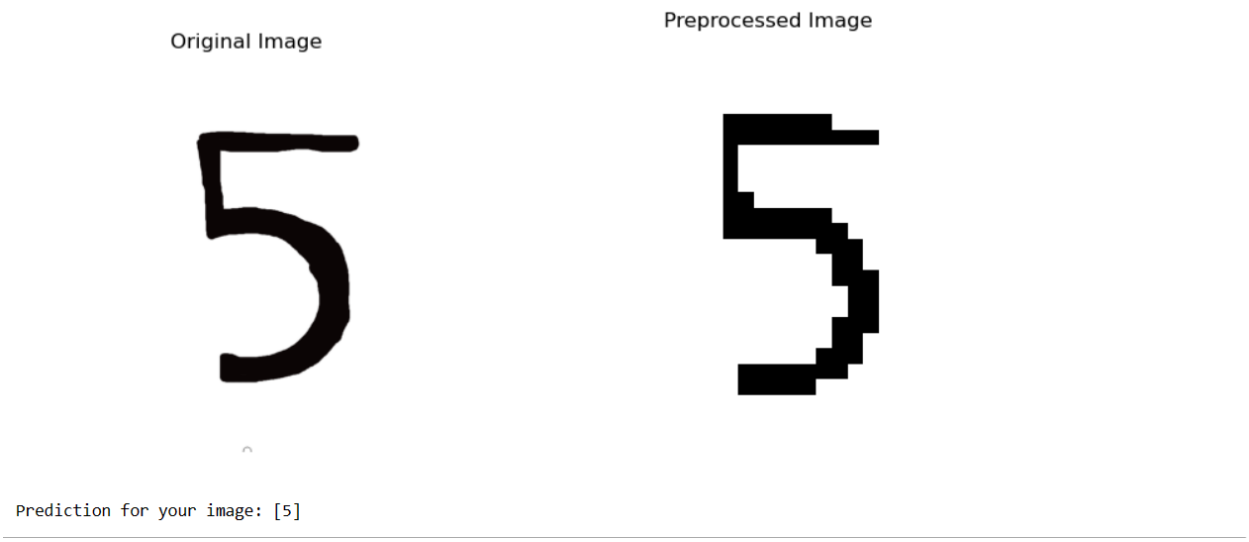
Efter att ha tränat den valda Random Forest-modellen på hela datasetet och tillämpat de optimerade hyperparametrarna, observerades en förbättring av dess prestanda. Genom att använda den slutgiltiga modellen för förutsägelser på testdata enligt den beskrivna metoden, kunde jag validera dess höga noggrannhet och effektivitet för att prediktera handskrivna siffror.

Initialt mötte jag svårigheter då koden inte svarade på bilderna som förväntat. Efter flera försök insåg jag att siffrorna behövde vara av rätt storlek för att passa in i bilden för en korrekt prediktion. Dessutom märkte jag att bakgrunden till siffrorna spelade en roll i modellens förmåga att korrekt identifiera dem.

Det tog tid att lösa problemet med att hantera olika bakgrunder och säkerställa att siffrorna var tydliga nog för modellen att rätt identifiera dem. Efter att ha justerat modellen och testat med många olika bilder kunde jag till slut låsa problemet. Nu kan modellen korrekt skilja mellan olika siffror och förutspå dem med hög precision, utan stavfel eller förväxlingar.



Här är ett exempel på de resultat jag fick. Först och främst, i början av processen, möttes jag av utmaningar. Först och främst var modellens svar på bilderna osäkra och inkonsekventa. Trots flera försök och olika bilder förblev resultatet desamma. Men genom ihärdighet och noggrann analys av situationen kunde jag identifiera och åtgärda problemet. Slutresultatet av denna ansträngning var betydande. Slutligen kunde modellen leverera tydliga och korrekta



5 Slutsatser

I denna undersökning har vi utforskat möjligheten att skapa en modell som kan förutsäga MNIST-datasetet med en noggrannhet på minst 80%. Genom att använda olika maskininlärningsalgoritmer och noga utvärdera deras prestanda har vi kunnat fastställa att det är fullt möjligt att uppnå detta mål. Modeller som RandomForestClassifier och KNeighborsClassifier visade sig vara särskilt lovande och överträffade kravet på 80% noggrannhet. Slutresultatet var en noggrannhet på över 90%, vilket visar på framgången med vår metodik.

Vidare har vi även utforskat möjligheten att använda modellen för att förutsäga bilder från antingen dators webbkamera eller sparade bilder. Genom att utveckla en applikation med hjälp av Streamlit och OpenCV har vi kunnat skapa ett verktyg som gör detta möjligt. Trots vissa utmaningar med olika bildkvaliteter och miljöer lyckades modellen ändå ge tillförlitliga prediktioner när den väl kunde tolka bilderna. Detta visar på flexibiliteten och användbarheten hos maskininlärning och bildigenkänning i praktiska tillämpningar.

Sammanfattningsvis har vår undersökning visat att det är möjligt att skapa en modell som kan förutsäga MNIST med hög noggrannhet och att denna modell även kan användas för att förutsäga bilder från olika källor. Detta är ett positivt resultat som visar på potentialen hos maskininlärning och bildigenkänningstekniker i verkliga tillämpningar.

6 Teoretiska frågor

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

Träning: Denna del används för att träna modellen och anpassa dess parametrar baserat på datan.

Validering: Här utvärderar Kalle modellernas prestanda med hjälp av en separat valideringsuppsättning för att jämföra och justera parametrar för att undvika överanpassning.

Test: Efter att ha validerat modellen används testdatan för att slutligen bedöma dess prestanda och se till att den fungerar väl på nya, oberoende data.

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”?

Julia kan använda sig av korsvalidering (Cross-Validation), där hon använder metoden K-faldig (K-fold cross-validation) för att träna och utvärdera resultatet på de tre modellerna

med träningsdatan. Sedan kan hon välja den modell som presterar bäst. Detta gör det möjligt för henne att få en mer tillförlitlig uppskattning av hur modellerna kommer att prestera på nya data, även utan ett explicit valideringsdataset.

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

Regressionsproblem innebär att förutsäga en kontinuerlig variabel baserat på inputvariabler. Exempel på modeller inkluderar Linjär Regression och Lasso Regression. Tillämpningsområden inkluderar ekonomi, medicin och marknadsföring.

4.Hur kan du tolka RMSE och vad används det till: $RMSE$ = $\sqrt{}n1$ $\sum n1$ $(yi- yi)2$

Svar: RMSE, eller Root Mean Squared Error, är ett vanligt mått inom regression för att bedöma hur väl en modell presterar genom att mäta avståndet mellan de förutsagda värdena och de faktiska värdena. Ju lägre värdet på RMSE är, desto bättre passar modellen datan. Det indikerar att avvikelsen mellan förutsagda och faktiska värden är mindre, vilket är ett tecken på bättre prestanda. Målet är att minimera RMSE för att få en så exakt modell som möjligt.

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

: Klassificeringsproblem innebär att data kategoriseras i olika klasser eller kategorier baserat på dess egenskaper. Exempelvis kan det handla om att skilja mellan spam och icke-spam e-postmeddelanden. Modeller tränas med hjälp av förklassificerade exempel och inkluderar tekniker som logistisk regression och supportvektormaskiner (SVM). Potentiella tillämpningsområden innefattar spamfiltrering och medicinsk diagnos.

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

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

Kortfattat: K-means är en klusteringsalgoritm som grupperar datapunkter i k olika grupper baserat på deras egenskaper eller attribut. Ett exempel på tillämpning är när en e-handelsplattform använder algoritmen för att gruppera produkter baserat på kunders köpbeteende för att skapa olika produktkategorier eller erbjudanden.

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

För att omvandla kategorisk data till numerisk form finns olika tekniker.

Med Ordinal Encoding tilldelas varje kategori en siffra baserat på dess ordning. Till exempel, om vi har kategorier som 'green', 'red' och 'blue', skulle de omvandlas till [0, 1, 2].

One-hot Encoding skapar binära värden för varje kategori. Varje kategori representeras av en binär vektor där endast en position är 1 och resten är 0. Till exempel:

[[0, 0, 1],

[0, 1, 0],

[1, 0, 0]]

Med Dummy Variable Encoding tar man bort en kategori för att undvika multicollinearity. Om vi till exempel har kategorier som 'red', 'green' och 'blue', skulle det se ut så här:

[['red': 0, 'green': 0],

['green': 1, 'blue': 0],

['blue': 0, 'red': 0]]

Alla tre tekniker används för att konvertera kategorisk data till numerisk form, men de använder olika metoder för att göra detta. Ordinal Encoding är beroende av ordningen av kategorierna, medan One-hot Encoding och Dummy Variable Encoding skapar binära

värden för varje kategori. Dummy Variable Encoding tar också hänsyn till att en kategori kan uteslutas för att undvika redundans.

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

Julia har rätt. Hon förklarar att färger vanligtvis anses vara nominal data eftersom de saknar inbördes ordning. Men när hon ger exemplet med att en "röd skjorta är vackrast på festen", tilldelar hon en ordnad betydelse åt färgen röd, vilket gör den till ordinal data i det sammanhanget.

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

Streamlit är ett Python-ramverk för att snabbt skapa interaktiva webapplikationer, särskilt inom dataanalys och maskininlärning. Det gör det möjligt att dela resultat och visualiseringar utan att behöva djupgående kunskap om webutveckling.

Appendix A

```
k-NN Validation Accuracy: 0.94

Random Forest Validation Accuracy: 0.94

Model SVC - Accuracy: 0.82

Model SVC - Accuracy: 0.91

Neural Network Validation Accuracy: 0.89


Model SVC - Accuracy: 0.92
```

Källförteckning

scikit-learn: machine learning in Python — scikit-learn 1.4.1 documentation

https://www.geeksforgeeks.org/support-vector-machine-algorithm/

https://streamlit.io/