Kunskapskontroll 2

Bildklassificering

Emil Nilsson

EC Utbildning

Maskininlärning - Kunskapskontroll 2

2025-03

Innehållsförteckning

1 Inledning

Bildklassificering är en viktig del inom AI och har tillämpningar i många olika yrken allt från fakturaskanning, handskriftsigenkänning och medicinska analyser. Därför är det viktigt att kunna träna modeller som effektivt kan utföra bildklassificering med hög noggrannhet.

MNIST innehåller 70,000 handskrivna siffror vilket gör det till ett bra dataset för att träna olika modeller och jämföra samt kontrollera prestanda.

Syftet med denna rapport är att se hur bra olika bildklassificeringsmodeller kan prestera på de olika handskrivna siffrorna i MNIST datasetet. För detta kommer jag specifikt att utvärdera 2 olika modeller. Multinomial Logistisk Regression och Random Forest. För att uppfylla syftet kommer jag ställa följande frågeställning:

- "Hur bra presterar Random Forest på MNIST-datasetet, och hur står den sig i jämförelse med en enklare modell (logistisk regression)?"

2 Teori

2.1 Bildklassificering och Maskininlärning

Bildklassificering är en viktig tillämpning av maskininlärning där en modell tränas för att känna igen och kategorisera bilder baserat på deras innehåll. Detta används inom en mängd olika områden, exempelvis inom medicinsk bildanalys, fakturaskanning och objektigenkänning i självkörande bilar.

I denna studie används maskininlärning för att klassificera handskrivna siffror från MNIST-datasetet. Modellerna som utvärderas är Multinomial Logistisk Regression och Random Forest.

2.2 Multinomial Logistisk Regression

Multinomial logistisk regression är en klassificeringsmodell som används för att förutsäga sannolikheten att en observation tillhör en viss klass. Den fungerar genom att använda softmax-funktionen, som omvandlar resultaten från en linjär funktion till sannolikheter för varje klass. Modellen väljer sedan den klass med högst sannolikhet. För mer information, se Scikit-learn dokumentationen (Scikit-learn, 2024) (1)

2.3 Random Forest

Random Forest är en ensemblemodell som kombinerar flera beslutsträd för att förbättra klassificeringsprestandan och minska risken för överanpassning. Varje beslutsträd i skogen tränas på en slumpmässig del av datasetet, och vid klassificering görs en röstning mellan träden där den mest valda klassen blir modellens slutgiltiga förutsägelse. För en mer detaljerad beskrivning av Random Forest, se Scikit-learns dokumentation. (2)

3 Metod

3.1 Dataset

I denna studie används MNIST-datasetet, som innehåller 70 000 bilder av handskrivna siffror (0–9) i upplösningen 28x28 pixlar. Datan delas in i två delar:

- Träningsdata: 56 000 bilder (80 %)
- Testdata: 14 000 bilder (20 %)

Varje bild omvandlas från en 28x28 matris till en 1D-vektor med 784 värden, vilket gör det möjligt att använda modeller som arbetar med vektorinmatningar.

3.2 Databehandling

För att säkerställa att de inmatade bilderna är kompatibla med MNIST-datasetet används en förbehandlingsfunktion (`preprocess_image`). Funktionen utför följande steg:

1. Gråskala – Omvandlar bilden till svartvit (gråskala) för att matcha MNIST-formatet.
2. Invertering – MNIST har svarta siffror på vit bakgrund, därför inverteras bilden om nödvändigt.
3. Binärisering – Bilden trösklas med Otsu's metod för att förstärka siffrans konturer.
4. Förstärkning av siffran – En morfologisk operation (dilation) tillämpas för att göra linjerna tydligare.
5. Bounding Box – Identifierar och beskär siffran för att ta bort onödig bakgrund.
6. Skalning till 20x20 pixlar – Anpassar siffrans storlek till 20x20 pixlar.
7. Placering i 28x28 bild – Justerar så att den bearbetade siffran matchar MNIST-formatet.
8. Centrering – Flyttar siffran så att den är korrekt placerad i mitten av bilden.
9. Normalisering – Omvandlar pixelvärdena till ett intervall mellan 0 och 1 för att förbättra modellens inlärning.

Denna preprocessering säkerställer att modellen får en enhetlig och optimerad inmatning, vilket förbättrar klassificeringsnoggrannheten.

3.3 Modeller och Träning (Scikit-learn, 2024)

3.3.1 Multinomial Logistisk Regression

- Solver: lbfgs – en numerisk optimeringsmetod för flervalsklassificering.
- multi_class: multinomial – modellen tränas med softmax-regression.
- max_iter: 1000 – antalet iterationer.
- **n**_jobs: -1 – använder alla processorkärnor för att snabba upp träningen.

3.3.2 Random Forest

- N_estimators: 100 (hittat via GridSearchCV, även om 400 eller 500 gav marginellt högre accuracy men ökade filstorleken avsevärt.

- Max_depth: 40 (hittat via GridSearchCV) – träden tillåts växa tills alla löv är rena eller har färre än min_samples_split datapunkter.
- N_jobs: -1 för att använda alla processkärnor
- Hyperparametersoptimering: GridSearch användes för att hitta de optimala hyperparametrarna

3.4 Utvärdering av modeller

För att säkerställa att modellerna presterar stabilt och generaliserar väl används cross-validation (korsvalidering). Cross-validation innebär att datasetet delas upp i flera delar, där modellen tränas på vissa delar och testas på andra, vilket ger en mer robust utvärdering.

I denna studie används 5-fold cross-validation för att utvärdera både Random Forest och Multinomial Logistisk Regression:

Denna metod delar upp träningsdatan i 5 olika delmängder, där modellen tränas på 4 delar och testas på den femte. Processen upprepas 5 gånger, och medelvärdet av accuracy beräknas för att få en mer tillförlitlig prestandaindikator. Modellerna utvärderas genom följande metoder (Scikit-learn, 2024):

1. Accuracy – Andelen korrekt klassificerade siffror. (3)
2. Cross-validation – Används för att säkerställa att modellerna presterar konsekvent över flera datadelningar. (4)

3.4.1 Sammanfattning:

Steg	Beskrivning
Dataset	MNIST, 70,000 bilder, 80/20 träning/test
Databehandling	Omvandling till 1D-vektorer, normalisering
Modeller	Multinomial Logistisk Regression och Random Forest
Hyperparameterar	Optimerade med GridSearchCV
Utvärdering	Accuracy och Cross-validation

4 Resultat och Diskussion

Accuracy för olika modeller	
Multinomial Logistisk Regression	91,99%

Random Forest	96,70%
Random Forest (GridSearchCV)	96,75%

4.1.1 Jämförelse av modeller

Resultaten visar att Random Forest presterade betydligt bättre än Multinomial Logistisk Regression, med en förbättring på nästan 5 procentenheter. Den optimerade versionen av Random Forest med GridSearchCV gav en ytterligare förbättring, men skillnaden mellan standardmodellen (96,70%) och den optimerade versionen (96,75%) var minimal.

Trots att GridSearchCV hjälpte till att identifiera optimala hyperparametrar, var förbättringen marginell, vilket väcker frågan om den ökade beräkningstiden var värd den lilla prestandaförbättringen.

4.1.2 Analys av metodval

- Preprocesseringen av bilder: Eftersom bilderna förbehandlades noggrant (gråskalekonvertering, binärisering, storleksändring och centrering), fick modellerna en enhetlig inmatning, vilket sannolikt förbättrade resultaten.
- Cross-validation: Genom att använda 5-fold cross-validation säkerställdes att modellerna generaliserade väl till nya data och inte bara presterade bra på träningsdatan.

5 Slutsatser

Studien syftade till att utvärdera vilken modell som presterade bäst på MNIST-datasetet.

Resultaten visade att Random Forest med GridSearchCV uppnådde den högsta accuracy på 96,75%, medan standard Random Forest fick 96,70% och Multinomial Logistisk Regression hamnade på 91,99%. Även om Random Forest med GridSearchCV presterade bäst, var förbättringen jämfört med standard Random Forest minimal (endast 0,05 procentenheter). Detta väcker frågan om den ökade beräkningstiden var värd den marginella förbättringen. Denna studie visar att även om hyperparameteroptimering kan förbättra prestandan, kan den ökade beräkningstiden vara en nackdel om förbättringen är marginell.

Vidare arbete skulle kunna inkludera att testa andra modeller såsom CNN, eftersom dessa ofta presterar bättre på bildklassificeringsuppgifter. Dessutom kan ytterligare hyperparameteroptimering utföras för att avgöra om ytterligare prestandaförbättringar kan uppnås., samt att optimera fler hyperparametrar för att se om ytterligare förbättringar är möjliga.

Denna analys ger insikter i hur väl olika modeller presterar på MNIST-datasetet och visar att Random Forest är en stark modell för bildklassificering även utan avancerad neurala nätverksarkitekturer.

6 Teoretiska frågor

6.1 Fråga 1:

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

- Träningsdata - Används för att träna modeller genom att justera parametrar för att hitta mönster i data.
- Valideringsdata - Används för att utvärdera, justera hyperparametrar och förbättra prestandan på modeller.
- Testdata - Används för att utvärdera modeller på osedda data och ger en bedömning på prestanda inför produktion.

6.2 Fråga 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”?

- Detta kan göras med Cross-validation

6.3 Fråga 3:

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

6.3.1 Exempel:

- Regressionproblem – När man vill förutspå ett kontinuerligt värde genom att analysera sambandet mellan den beroende variabeln och en eller flera oberoende variabler.

- Linjär regression – Enkel och flervariabel regression för att modellera samband mellan en beroende variabel och en eller flera oberoende variabler.

- Random Forest Regressor – En ensemblemodell som använder flera beslutsträd för att göra robusta förutsägelser.

6.3.2 Tillämpningsområden:

- Finans - Förutspå aktiekurser baserat på historiska data.
- Fastigheter - Utvärdering av huspris baserat på t. ex antal rum, läge och storlek.

6.4 Fråga 4:

Hur kan du tolka RMSE och vad används det till:

6.4.1 Tolkning:

- RMSE – Medelfelet i samma enhet som den beroende variabeln

6.4.2 Användning:

- Används för att jämföra olika modellers noggrannhet
- Visar hur långt ifrån de verkliga värdena modellerna tycks vara.

6.5 Fråga 5:

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

6.5.1 Klassificeringsproblem:

- Uppstår när man vill förutsäga vilken klass en observation tillhör.

6.5.2 Modeller:

- Logistisk regression – En modell för enkel binär klassificering.
- Random Forest – En ensemblemodell som kombinerar flera beslutsträd

6.5.3 Confusion Matrix:

- En tabell som visar hur bra en klassificeringsmodell presterar.
- Jämför faktiska värden med prediktionerna, hur rätt ofta den predikterar rätt.

6.6 Fråga 6:

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

6.6.1 K-Means:

- Klustringsalgoritm som används för att gruppera datapunkter i olika kluster.

6.6.2 Tillämpningsområde:

- Marknadsföring - Kundsegmentering, för att dela in kunder i olika grupper för beteende analys.

6.7 Fråga 7:

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

6.7.1 Ordinal Encoding:

- Används när kategoriska variabler har en viss ordning.
- Omvandlar kategoriska värden till numeriska genom att tilldela varje kategori ett unikt nummer.

6.7.2 Exempel:

- Olika klädstorlekar, small, medium och large.

Storlek:	Ordinal Encoding
Small	0
Medium	1
Large	2

6.7.3 One-Hot encoding:

- Skapar binära kolumner för varje kategori. Varje kategori representeras av en egen kolumn där endast en kolumn har värdet 1 (de andra är 0).

6.7.4 Exempel:

Färg	Röd	Blå	Grön
Röd	1	0	0
Blå	0	1	0
Grön	0	0	1

6.7.5 Dummy Variable Encoding:

- Dummy Encoding är en variant av One-Hot Encoding där en kategori utelämnas för att undvika multikollinearitet (dvs. att en variabel blir en linjär kombination av de andra)

6.7.6 Exempel:

Färg	Röd	Blå
Röd	1	0
Blå	0	1
Grön	0	0

6.8 Fråga 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?

6.8.1 Svar:

- Göran har rätt i att påstå att data antingen är ordinal eller nominal.
- Julia har också rätt i att säga att det måste tolkas.
- Det beror på sammanhanget och hur datan skall användas.

6.9 Fråga 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?

6.9.1 Vad är Streamlit?

- Streamlit är ett Python bibliotek som används för att snabbt kunna skapa applikationer för dataanalys och maskininlärning.

6.9.2 Användningsområden:

- Dashboards – Visualisera dataframes, skapa interaktiva grafer och diagram för dataanalys.

- Maskininlärning - Ladda upp ML modeller för realtidsprediktioner.

7 Självutvärdering

1. Utmaningar du haft under arbetet samt hur du hanterat dem.
   Min största utmaning var definitivt att hitta det jag ville i cv2 biblioteket.

2. Vilket betyg du anser att du skall ha och varför.
   Jag skulle säga att jag ligger mellan G och VG. Det hänger på hur väl jag gjort min rapport.
   Koden är jag väldigt nöjd med.

3. Något du vill lyfta fram till Antonio?
   Väldigt bra och rolig kurs, definitivt den roligaste hittills!

Appendix A

https://github.com/2emiln/Machine-Learning-Practice

Källförteckning

(1) https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html

(2) https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html

(3) https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.cross_val_score.html

(4) https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.cross_validate.html