Maskin Inlärning



Jesper Kokkonen

EC Utbildning

Kunskapskontroll 2

2024–03

# Abstract

This study focuses on the development of a machine learning model with the aim to predict the right handwritten digit with a high accuracy using the MNIST dataset. I’ll be testing 4 different machine learning models and test them to see which one is the most accurate on the MNIST dataset.

**Skapas automatiskt i Word genom att gå till Referenser > Innehållsförteckning.**

Innehållsförteckning

[Abstract 2](#_Toc162010412)

[1 Inledning 1](#_Toc162010413)

[1.1 Streamlit 1](#_Toc162010414)

[2 Teori 2](#_Toc162010415)

[2.1.1 Descision Tree Classifier 2](#_Toc162010416)

[2.1.2 Random Forest Classifier 2](#_Toc162010417)

[2.1.3 Linear Support Vector Machine 2](#_Toc162010418)

[2.1.4 Support Vector Machine 2](#_Toc162010419)

[3 Metod 3](#_Toc162010420)

[3.1 Databas 3](#_Toc162010421)

[3.2 Standard Scaler 3](#_Toc162010422)

[3.3 Testandet av modeller 3](#_Toc162010423)

[3.4 Val av modell 3](#_Toc162010424)

[3.5 OpenCV och Streamlit 3](#_Toc162010425)

[4 Resultat och Diskussion 4](#_Toc162010426)

[4.1 Test av modeller 4](#_Toc162010427)

[4.2 Test av webkamerabild 5](#_Toc162010428)

[5 Slutsatser 6](#_Toc162010429)

[6 Teoretiska frågor 7](#_Toc162010430)

[7 Självutvärdering 9](#_Toc162010431)

[Appendix A 10](#_Toc162010432)

[Källförteckning 11](#_Toc162010433)

# Inledning

Syftet med denna rapport är att skapa en maskininlärningsmodell för att hitta den mest träffsäkra modellen av de fyra valda och för att uppfylla syftet så kommer följande frågeställning(ar) att besvaras:

1. Kan man skapa en modell som får minst 90% träffsäkerhet?

## Streamlit

För att kunna skapa en applikation så användes Streamlits och OpenCV så att VG-frågan kan besvaras.

# Teori

### Descision Tree Classifier

Decision Trees är mångsidiga maskininlärningsalgoritmer som kan utföra både klassificerings- och regressionsuppgifter och till och med flerutgångsuppgifter. De är kraftfulla algoritmer som kan anpassa komplexa datauppsättningar. (Géron, 2019, p. 175).

### Random Forest Classifier

Random Forest Classifier är en ensemble av Decision Trees. Random Forest-algoritmen introducerar extra slumpmässighet vid odling av träd, istället för att söka efter den allra bästa funktionen när den delar upp en nod, söker den efter den bästa funktionen bland en slumpmässig delmängd av funktioner. (Géron, 2019, p. 197).

### Linear Support Vector Machine

En Support Vector Machine (SVM) är en kraftfull och mångsidig maskininlärningsmodell, som kan utföra linjär eller olinjär klassificering, regression och avvikande detektering. SVM är väl lämpade för klassificering av komplexa datamängder som är små eller medelstora. (Géron, 2019, p. 153).

### Support Vector Machine

Även om linjära SVM-klassificerare är effektiva och fungerar förvånansvärt bra i många fall, är många datauppsättningar inte ens i närheten av att vara linjärt separerbara. Ett tillvägagångssätt för att hantera icke-linjära datauppsättningar är att lägga till fler funktioner, såsom polynomegenskaper; i vissa fall kan detta resultera i en linjärt separerbar datauppsättning. (Géron, 2019, p. 157).

# Metod

## Databas

Databasen som användes var Open Machine Learnings MNIST, databasen har 70 000 handskrivna siffror. Scikit-learns ramverk användes för att ladda ner MNIST-databasen för testning.

## Standard Scaler

Innan något test kördes standardiserades data med hjälp av StandardScaler vilket betyder att datauppsättningen omvandlas och fördelningen får ett medelvärde på 0 och standardavvikelse 1

## Testandet av modeller

Vid testet användes 14 000 siffror för att snabbaupp modellernas klassificering eftersom om data inte skalas ner kommer testningen att alldeles ta väldigt lång tid. Efter varje modell så predikterades den och lades upp på en confusion matrix för att få en uppfattning om vilken modell som presterade bäst.

## Val av modell

Valet av modell blev SVC som presterade bättre än Dicision Tree, Random Forest Classifier och LinearSVC på accuracy score. SVC modellen använde GridSearchCV med hyperparametrarna ”C” och ”Gamma”och kernel som använde var ”poly”.

## OpenCV och Streamlit

För att ge sig på VG uppgiften användes OpenCV (Open Source Computer Vision Library) vilket är ett bibliotek med öppen källkod för datorvision och maskinlärning. OpenCV användes för att kunna visualisera bilderna som har laddas upp eller tagits med webkamera. Streamlit är ett Python bibliotek med öppen källkod som användes för att skapa en enkel webapplikation där uppladdades min modell samt valet att ladda upp egen bild eller att ta en bild med webkamera.

# Resultat och Diskussion

I detta kapitel kommer jag gå igenom och diskutera resultaten av de olika modeller som användes i denna uppgift.

## Test av modeller

En bild som visar skärmbild, Färggrann, kvadrat, text

Automatiskt genererad beskrivningEn bild som visar skärmbild, text, Färggrann, kvadrat

Automatiskt genererad beskrivning

En bild som visar text, skärmbild, Färggrann, diagram

Automatiskt genererad beskrivningEn bild som visar text, skärmbild, diagram, Färggrann

Automatiskt genererad beskrivning

Av de fyra valda modellerna presterade SVC marginellt bättre än Random Forest Classifier. Här skulle det förmodligen gå att få ett lite bättre resultat om så var fallet men i pratar om så liten skillnad att det inte skulle vara anmärkningsvärt.

## Test av webkamerabild

När det kommer till att prediktera bilder som var tagna i Streamlit applikationen var resultatet inte så vidare bra av det tio olika siffror som togs var det bara sex av tio som predikterades korrekt.

Detta har med störst sannolikhet att göra med upplösning, ljus och kvaliteten på kameran.

När jag laddade upp bilder skrivna i paint så blev resultatet bättre modellen predikterade åtta av tio korrekt. Detta ledde mig till mitt antagande att kvaliteten på webkamerabilderna inte var något vidare bra.

En bild som visar skärmbild, text, Multimedieprogram, programvara

Automatiskt genererad beskrivning En bild som visar text, skärmbild, programvara, klocka

Automatiskt genererad beskrivning

Figur 1: Webkamerabild Figur 2: Skriven i paint

# Slutsatser

Frågeställningarna lydde ”Kan man skapa en modell som får minst 90% träffsäkerhet?” och svaret på den frågeställning är ja. Både SVC och Random Forest Classifier lyckades få minst en 90% träffsäkerhet. Att man kan uppnå hela 96% på en SVC modell utan mycket kunskap om maskininlärning på detta dataupplägg var förvånade och inte något som jag hade förväntat mig.

# Teoretiska frågor

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

Tränings delen tränar man data på. Validering utvärderas modellerna och väljer den bästa.

Test testas den valda modellen för att få en uppskattning på hur bra modellen kommer att fungera på ny 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”?

Hon kan använda k-fold cross validation, för att sen se vilken modell som är bäst. Hon kan även göra en Gridsearch på hyperparametrarna med GridsearchCV. Sen söker hon igenom för att hitta den kombination som ger bäst cross validation score efter detta kan hon välja modell.

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

Ett regressionsproblem är en typ av övervakat inlärningsproblem inom maskininlärning, där målet är att förutsäga ett kontinuerligt numeriskt värde baserat på en eller flera inputfeatures.

Linjär Regression, Support Vector Machines (SVM), Beslutsträd. Ett potentiellt tillämpningsområde är att titta på inkomst och ålder

4. Hur kan du tolka RMSE och vad används det till: 𝑅𝑀𝑆𝐸 = √∑(𝑦𝑖 − 𝑦̂𝑖 ) 2 𝑖

RMSE kan vi tolka som vår prediktioners medelavstånd till de sanna värdena.

RMSE används till när du vill utvärdera din validerings data eller test data

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

Problem där målkolumnen är en kategorisk kolumn kallas klassificeringsproblem. Binära klassificeringsproblem har två möjliga kategorier, t.ex. ja eller nej, medan problem med flerklassiga klassificeringar har fler än två möjliga kategorier.

Modell exempel Logistisk regression.

Kommer kunden ”churna”? Kommer patienten överleva? Är mejlet spam/inte spam? Är det en man/kvinna på bilden?

Confusion Matrix är en specifik tabelllayout som möjliggör visualisering av prestanda för en algoritm

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

K-means algoritmen försöker hitta varje klusters centrum (centroid observation). Därefter så ”klassificerar man” alla punkter till det kluster vars centrum är närmast. Den kan tillämpas när man exempelvis vi vill klassificerar blommor (Iris data set).

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

Ordinal encoding: varje unik kategori har ett numeriskt värde

One-hot encoding: Binära variabler för varje kategori. Is\_cat1=0 eller 1 osv

Dummy variable: Binära kategorisering med en mindre än antal kategorier. Sista blir alltid nollor eftersom den inte är någon av de tidigare kategorierna och slutledningsförmåga.

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?

Julia har rätt det måste tolkas. Som i hennes exempel kan färgerna inte har någon inbördes ordning vilket är nominal men om färgerna har en inbördes såsom att röd är den vackraste färgen så är den ordinal.

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 ett ramverk med öppen källkod för att skapa dataapplikationer i Python för Machine Learning och Data Science.

# Självutvärdering

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

De utmaningarna jag har haft under arbete var VG delen och rapporten. Det tog väldigt lång tid innan jag hittade relevant information om vad för någon kod man skulle kunna använda för att laddauppbilderna och hur man omvandlar bilderna via applikationen. Jag hanterade problemet för de mesta genom att googla runt för att kunna hitta relevant information, men även underlektionstid så lärde jag mig en del om hur jag skulle kunna gå till väga för att lösa VG delen. Rapporten delen tyckte jag var svår att skriva till mesta del har det nog med att jag inte är så vidare van vid att skriva rapporter och att VG delen tog en så långtid att göra.

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

Jag anser att jag ska ha betyget Godkänt till mesta dels för att jag tycker att jag har uppnått de krav som ställdes för att få det betyget. Jag gjorde VG delen ihop om att få Väl Godkänt i betyg men jag tycker inte min rapport når upp till det kravet som jag anser är av Väl Godkänt standard.

1. Något du vill lyfta fram till Antonio?

Nej.

# Appendix A

# Källförteckning

Géron A. (2019): Hands-On Machine Learning with Scikit-Learn, Keras & TensorFlow

<https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.StandardScaler.html>

<https://opencv.org/about/>

<https://streamlit.io/>