Maskininlärning

Modellering av MNIST



My Gieng

EC Utbildning

Kunskapskontroll 2 – Maskininlärning

202503

# Abstract

The purpose of this study is to understand the different concepts and components of machinelearning and to apply the knowledge to model MNIST dataset. Random Forest Classification model performed the best, based on accuracy score.

Innehållsförteckning

[Abstract 2](#_Toc193493367)

[1. Inledning 1](#_Toc193493368)

[1.1. Maskininlärning 1](#_Toc193493369)

[1.2. MNIST 1](#_Toc193493370)

[1.3. Syfte 1](#_Toc193493371)

[2. Teori 2](#_Toc193493372)

[2.1 Klassificeringssmodeller 2](#_Toc193493373)

[2.1.1 Support Vector Machines 2](#_Toc193493374)

[2.1.2 Random Forest Classification 2](#_Toc193493375)

[2.2 Grid Search 2](#_Toc193493377)

[2.3 Prestandamått 2](#_Toc193493378)

[2.3.1 Accuracy 2](#_Toc193493379)

[2.3.2 Precision och Recall 3](#_Toc193493380)

[3. Metod 4](#_Toc193493381)

[4. Resultat 5](#_Toc193493382)

[5. Slutsatser och Diskussion 6](#_Toc193493383)

[6. Teoretiska frågor 7](#_Toc193493384)

[7. Självutvärdering 10](#_Toc193493385)

[Källförteckning 11](#_Toc193493386)

# Inledning

## Maskininlärning

Maskininlärning innebär att programmera datorer för att kunna lära sig från data utan att explicit vara programmerade. Det går till genom att lära systemet via ett tränings set, där varje tränings set delas in i olika tränings instanser. Maskininlärning är bra för att man låter datorn förenkla kod och därav prestera bättre jämfört med den traditionella metoden, dessutom kan datorn, genom maskininlärning, hitta mönster för att kunna göra förutsägelser.

Maskininlärning kan bland annat delas in i väglett- och icke väglett lärande. Ett väglett lärande innebär att träningsdata som matas in i modellen är de önskade lösningarna för att därefter kunna förutsäga nya data. Inom väglett lärande kan uppgifter/problem som klassificering och regression utföras. Vid ett icke väglett lärande innehåller träningsdata inte lösningarna eftersom systemet försöker lära av sig själv eller hitta samband i data. För att ett system ska fungera bra är det viktigt att träningsdata inte är för liten eller inte är representativ.

För att kunna utvärdera hur bra en modell presterar är att låta den tränas med olika data. Vanligtvis delas data upp i olika uppsättningar, träningsuppsättning och testuppsättning. Träningsdata används för att träna modellen medan med testdata uppnår man en uppskattning hur bra modellen kan prestera på ny osedd data.

Det finns olika maskininlärningsmodeller. Support Vector Machine är en kraftfull och mångsidig modell då den kan utföra både linjära och icke linjära klassificeringar och regressioner. I korthet går det ut på att hitta ett så bra sätt att separera data i olika klasser. Random forest är en annan kraftfull modell som är särskilt bra vid användning av både klassificering och regressions problem. Denna modell går istället ut på att använda beslutsträd för att göra prediktioner. (Geron 2019)

## MNIST

MNIST är en databas som ofta används inom maskininlärning (Geron 2019). Databasen består av handskrivna siffror från både gymnasieelever och anställda från ett företag. Databasen har en uppsättning av 60 000 träningsbilder och 10 000 testbilder av 28x28 pixlar (OpenML 2014)

## 1.3. Syfte

Syftet med denna rapport är att modellera MNIST databas genom ett komplett maskininlärningsflöde, där minst två olika modeller ska tränas och jämföras. Målet är att utvärdera vilken modell som presterar bäst gällande klassificering av varje bild till rätt siffra genom att använda olika prestandamått.

# Teori

## 2.1 Klassificeringssmodeller

Klassificeringsmodeller innebär att kunna förutsäga vilken klass ett objekt tillhör. Några modeller som kan hantera klassificeringsproblem är logistisk regression, support vector machines och random forest classification (Geron 2019).

### Support Vector Machines

#### Support Vector Machine är en kraftfull och mångsidig modell. I korthet går det ut på att hitta ett så bra sätt att separera data i olika klasser. Önskvärt inom support vector machine är att hitta en så bred avstånd mellan de olika klasserna som möjligt och samtidigt i den mån det går undvika överträdelse av olika data mellan separeringen. Inom support vector machine finns olika hyperparametrar, C är en av dessa. C bestämmer hur flexibel modellen ska vara för tillåtelse av överträdelse av data mellan separeringen. Support vector machine kan utföra både linjära och icke linjära klassificeringar. Linjär support vector machine går ut på att hitta en rak linje för separering. När en linjär separation inte är möjlig kan olika kernel användas, såsom Gaussian radial basis function kernel, som möjliggör en linjär separation för en icke linjär kernel. Detta sker med hjälp av hyperparametrar såsom C och gamma (γ) (Geron 2019).

### Random Forest Classification

### Random forest är en annan kraftfull modell som är särskilt bra vid användning av både klassificering och regressions problem. Denna modell går ut på att använda beslutsträd för att göra klassificeringar. För att skapa dessa beslutsträd används oftast en teknik som heter bagging. Hyperparametrar inom random forest är bland annat maximum depth som regulariserar djupet för beslutsträden (Geron 2019).

## 2.2 Grid Search

Grid search är en metod som gör det möjligt att hitta de bästa hyperparametrarna för varje modell. Detta för att optimera prestandan för varje modell samt för att undvika över- eller underfitting (Geron 2019).

## 2.3 Prestandamått

### 2.3.1 Accuracy

Accuracy är ett prestandamått för att utvärdera klassificeringsproblem. Detta är ett mått på andelen korrekt förutsägelser (Geron 2019).

# Precision och Recall

Precision mäter andel positiva förutsägelser som faktiskt är positiva (Geron 2019)..

Recall mäter andelen av den faktiska positiva som korrekt identifierades av modellen (Geron 2019)..

I praktiken är det oftast praktiskt att använda en kombination av precision och recall till ett mått som kallas för F1 score, som är ett medelvärde av precision och recall. En modell kommer endast få hög F1 score om både precision och recall är höga (Geron 2019).

# Metod

MNIST databasen hämtades via OpenML. Därefter hämtades en beskrivning av databasen för att få en bakgrundsinformation om vad databasen består av för att kunna bilda en uppfattning.

Databasen delades upp i två variabler, X och y, för att därefter delas in i tränings-, validering- och test set. Bilderna standardiserades med hjälp av StandardScaler.

De modeller som valdes för att kunna lösa klassificeringsproblemet var Random Forest och Support Vector Machine. Grid Search användes för optimering av hyperparametrarna för de valda modellerna. För Support Vector Machine optimerades parametrarna C och gamma och för Random Forest max depth.

Support Vector Machine tränades med standardiserade data då denna modell presterar bättre med standardiserade data medan med Random Forest tränades med icke standardiserade data. Detta för att Random Forest inte är lika känslig för standardisering av data då denna modell går ut på att hitta kluster för att hitta bästa separering av klasserna, det vill säga påverkar prestandan inte om data är strukturerad eller inte. För att undvika över- eller underanpassning användes cross validation där modellen får tränas i olika kombinationer.

När modellerna var tränade med hjälp av träningsdata användes validerinsdata för att utvärdera modellerna för att därefter kunna välja den modell som presterade bäst utifrån träningsdata. För att bedöma vilken modell som presterade bäst användes ett accuracy värde, som är ett mått för hur ofta modellen gör rätt förutsättningar.

Därefter utvärderades resultaten och en modell bestämdes för att utvärdera den bästa tränade modellen med en ny osedd data, testdata. Modellen som valdes ut refittades på tränings- och validerings data. En classification report utfördes för att få en mer djupgående bild över prestationen för den bästa modellen.

# Resultat

Resultatet baseras på utvärdering av modellerna Support vector machine och Random Forest. De olika accuracy värden ses i Tabell 1. Utifrån accuracy värden valdes den modell som presterades bäst, det vill säga Random forest.

|  |  |
| --- | --- |
| **Accuracy för de olika modellerna** | |
| Linjär Support Vector Machine | 0,9232 |
| Support Vector Machine radial basis function kernel | 0,1158 |
| Random Forest | 0,9690 |

Tabell 1: Accuracy för de valda modellerna

Resultatet vid körning av testdata vid Random Forest gav en accuracy på 0,9674, det vill säga något mindre än accuracy vid träningen av modellen. En Classification rapport för Random Forest ses i tabell 2.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Precision | Recall | F1 score | Support |
| 0 | 0,98 | 0,98 | 0,98 | 983 |
| 1 | 0,99 | 0,99 | 0,99 | 1152 |
| 2 | 0,94 | 0,97 | 0,96 | 967 |
| 3 | 0,96 | 0,95 | 0,95 | 1034 |
| 4 | 0,96 | 0,97 | 0,97 | 906 |
| 5 | 0,98 | 0,96 | 0,97 | 937 |
| 6 | 0,98 | 0,99 | 0,98 | 961 |
| 7 | 0,97 | 0,97 | 0,97 | 1055 |
| 8 | 0,96 | 0,95 | 0,96 | 969 |
| 9 | 0,96 | 0,94 | 0,95 | 1036 |
| Accuracy |  |  | 0,97 | 10000 |
| Macro avg | 0,97 | 0,97 | 0,97 | 10000 |
| Weighted avg | 0,97 | 0,97 | 0,97 | 10000 |

Tabell 2: Classification Report för Random Forest

# Slutsatser och Diskussion

Modellerna som tränades var Support Vector Machine och Random Forest. Support Vector Machine, linjär och radial basis function kernel, presterade sämre accuracy vid träningen jämfört med Random Forest, som resulterade i en accuracy på 91,71%, 11,58% respektive 96,9%. Detta visar att Random Forest är en bättre modell att använda vid klassificering av handskrivna siffror. Radial basis function presterade sämst, med en accuracy på 11,58%, vilket kan vara ett resultat av att det är för komplext modell.

Då Random Forest gav bäst accuracy användes den för att testa modellen. Vid testning av modellen resulterade det i en accuracy på 96,74%. Detta resultat är väldigt lik den som fick vid träning vilket är önskvärt eftersom detta innebär att modellen generaliserar bra samt fungerar väl vid ny osedd data.

I classification report kan resultatet tolkas genom att modellen har en bra prestation då F1 score är 0,97. För att få en hög F1 score behöver både recall och precision vara hög vilket innebär att modellen kan klassificera de handskrivna siffrorna väl.

Siffran noll, ett och sex är modellens lättaste handskrivna siffror att känna igen, då både recall och precision är höga för samtliga. Siffran två hade lägst precision men en relativt hög recall vilket innebär att modellen är bra på att hitta siffran men samtidigt klassificera den fel.

Resultatet visar att modellen presterar bra på MNIST databas med en accuracy på 97% samt att recall och precision är i stort sett höga för samtliga siffror. Slutsatsen som kan dras är att Random Forest är en effektiv modell för att klassificera handskrivna siffror.

# Teoretiska frågor

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

Svar: Träningsdata: används för att träna modellen, vilket innebär att modellen får lära sig mönstren och relationerna i data. I denna del anpassar modellens parameter för att minimera fel eller optimera prestandan.

Valideringsdata: används för att utvärdera och justera modellen. Därefter väljs den bästa modellen ut. Tex används valideringsdata för att se hur bra modellen presterar med ny data. Valideringsdatan används även för att modellen inte ska bli för anpassad efter träningsdatan och undviker därmed overfitting (som sker pga man har för komplexa modeller, tex polynomregression istället för linjär regression).

Testdata: används för att utvärdera modellens prestanda efter att ha tränats om med hjälp av tränings och valideringsdatan. Testdata har tidigare inte används utan det är först när modellen är vald som testdata används, detta för att ge en realistisk uppfattning om hur bra modellen kommer prestera på nya osedda data.

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 ”validerings dataset”?

Svar: För att kunna använda träningsdatan utan en valideringsdata kan Julia använda Cross Validation och för varje iteration som används med hjälp av cross validation kan Julia räkna ut RMSE och den modell som genererar lägst RMSE är den bästa modellen.

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

Svar: Regressionsproblem är att man vill hitta ett samband mellan en kontinuerlig variabel (Y) och oberoende variabel/ler (x)

Några modeller som används är linjär regression, Beslutsträd, Random Forest Regression, Support Vector Machines.

Potentiella tillämpningsområden är tex hur inkomst beror på ålder, huspriser beror på faktorer som antal rum, kvadratmeter.

Fråga 4: Hur kan du tolka RMSE och vad används det till: � � 𝑅𝑀𝑆𝐸 =√1 𝑛 ∑(𝑦𝑖 −𝑦̂𝑖)2 1

Svar: RMSE kan tolkas som våra prediktioners medelavstånd till de sanna värdena. RMSE används som ett mått på prestandan vid jämförelse av olika modeller. Lägst RMSE innebär oftast bättre modell.

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

Svar: Ett klassificeringsproblem innebär att ta reda på vilken klass den beroende variabeln kan anta baserat på ett eller flera oberoende variabler.

Exempel på modeller är beslutsträd, random forest classification och support vector machines.

Några tillämpningsområden är att undersöka bland annat inom vården om en patient kommer överleva eller inte, inom företag ifall en kund kommer lämna eller inte, om mejl är spam eller inte.

Confusion matrix är ett verktyg för att utvärdera klassificeringsmodeller.

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

Svar: är en del av unsupervised learing där data försöks grupperas i kluster/grupper. Användbart för företag för att undersöka olika kundgrupper och deras beteenden för att därefter anpassa marknadsföring till olika kundgrupper.

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

Svar: Ordinal encoding: Är ett sätt att transformera kategorisk data till numerisk data där det finns en rangordning mellan kategorierna (ordinala variabler), tex betyg. Resultat av ordinal encoding blir följande  
Låg = 0, Mellan = 1, Hög = 2

One-hot encoding: är ett sätt att transformera kategorisk data till nominella variabler (utan rangordning). Tex färger. Resultat av one-hot encoding blir en vektor/matris med följande resultat

Röd blir [1, 0, 0], Blå blir [0, 1, 0], Grön blir [0, 0, 1]

Dummy variable encoding: en variant av one hot encoding MEN där sista kolumnen inte innehåller några 1 värden). Med one hot encoding skapas 3 kolumner, en för varje kategori. Men om vi känner till 2 av dessa kolumner kan man enkelt räkna ut den tredje kolumnen och därav behövs ingen extra kolumn = sparar minne.

Låt oss säga att vi har följande resultat av one hot encoding:

Röd blir [1, 0, 0]

Blå blir [0, 1, 0]

Grön blir [0, 0, 1]

Med dummy variable encoding blir resultatet istället

Röd blir [ 0, 0]

Blå blir [0, 1, 0]

Grön blir [0, 0, 1]

Dvs med dummy variable encoding skapas C-1 kolumner istället för C kolumner. Detta medför att det blir mindre upprepning.

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?

Svar: Båda har delvis rätt då data antingen är ordinal eller nominal men för att data ska vara ordinal behövs en ”tolkning” göras då det finns en rangordning mellan kategorierna.

Precis som Julia säger kan färgerna som röd, grön och blå vara nominal utan inbördes rang mellan de men om en röd skjorta innebär vackraste färgen, blir det en ordinaldata då det helt plötsligt finns en rang mellan färgerna, i just det fest-sammanhanget.

Fråga 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?

Svar: Streamlit är ett dataprogram som gör det möjligt att skapa data applikationer, används för visualisering av data inom maskininlärning och data analys.

# Självutvärdering

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

Som tidigare är tiden alltid en utmaning för mig och för att hinna med det som är planerat är det viktigt att hålla tidsramen vilket inte alltid är lätt, speciellt när ämnet är främmande och komplext för en och det blir mycket självstudier.

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

Rätta objektivt.

1. Något du vill lyfta fram till Antonio?

-

# Källförteckning

Geron, Aurelien. (2019). *Hands-on Machine Learning with Scikit-Learn, Keras, and Tensor-Flow.* Canada: O’Reilly Media.

OpenML (2014). MNIST\_789. (Elektronisk). Tillgänglig: https://openml.org/search?type=data&status=active&id=554&sort=runs. (Läst 202-03-05).