Maskininlärning med MNIST

”The Hello World of Machine Learning”

Sebastian Strömberg

EC Utbildning

Kunskapskontroll Maskininlärning

2024-03

Abstract

This study explores the domain of machine learning using the MNIST-dataset, aiming to understand the best performing models and the machine learning workflow. Using three different models, Logistic Regression, Support Vector Machine and Random Forest Classifier. The findings concluded that there was not a clear winner in the end but that both SVM and RFC performed almost identically.

Förkortningar och Begrepp

AI – Artificiell Intelligens

Churna - En kund som slutar använda en tjänst eller produkt

EDA – Exploratary Data Analysis

Kernel – Används för att förvandla data till en högre dimension (Ex. 2D till 3D) så att man kan använda SVM till icke linjära problem

LG – Logistic Regression

MNIST – Ett dataset med handskrivna siffror mellan 0 till 9

One vs Rest – En metod för Logistic Regression

RFC – Random Forest Classifier

SVM – Support Vector Machine

Innehållsförteckning

1 Inledning

Dagens samhälle blir bara mer digitaliserat och massvis med data samlas in. Allt ifrån våra intressen, vanor, köphistorik lagras. Men vad kan man använda denna data till? Inom maskininlärning och AI kan vi lära modeller att prediktera när en kund kommer churna, om ett mejl du får är spam eller inte spam, kommer din aktie gå upp eller ner. Hela idén med en maskininlärningsmodell som kan prediktera saker är otroligt intressant.

I detta arbete kommer vi att ta en närmare titt på MNIST-datasetet som även kallas ”the hello world” av maskininlärning. Rapporten kommer att gå igenom hela maskininlärningsflödet från datahantering till färdig modell där följande frågeställningar kommer att besvaras.

1. Hur går maskininlärningsflödet till
2. Vilken modell presterar bäst på MNIST-datasetet

2 Teori

2.1 Modellutvärdering

Metoden som har använts för att utvärdera modellerna är Scikit-learns inbyggda metod Classification Report som mäter följande aspekter:

- Precision: Modellens positiva prediktioner som blivit rätt. Exempel, modellen predikterar att 100 patienter har cancer, 90 av dessa visade det sig hade cancer och 10 inte. Så precision blir 90/ 90+10 vilket ger oss en precision på 90%.

$$Precision = \frac{\text{Antal True Posiives}}{\text{Antal True Positives} + \text{Antal False Positives}}$$

- Recall: Modellens förmåga att korrekt hitta de positiva resultaten. Exempel, om vi har 100 bilder på hundar och modellen hittar 90 utav dom har vi en Recall på 90%.

$$\text{Recall} = \frac{Antal\ True\ Positives}{Antal\ True\ Positives + Antal\ False\ Negatives}$$

- F1-score: Mäter balansen mellan Precision och Recall.

$$\text{F1} = 2 \times \frac{Precision \times Recall}{\text{Precision} + \text{Recall}}$$

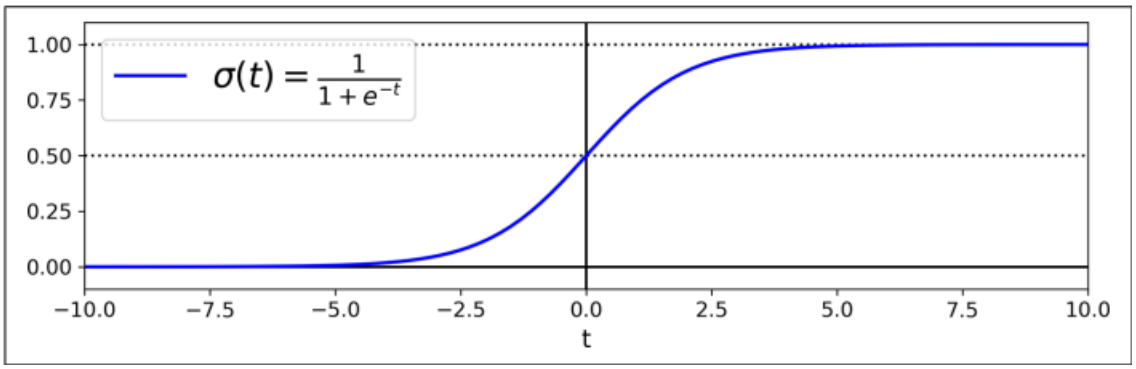
- Accuracy: Mäter andelen korrekta prediktioner i procent.

$$\text{Accuracy} = \frac{Antal\ Korrekta\ Prediktioner}{\text{Totalt Antal Prediktioner}}$$

2.2 Klassificeringsmodeller

2.2.1 Logistic Regression

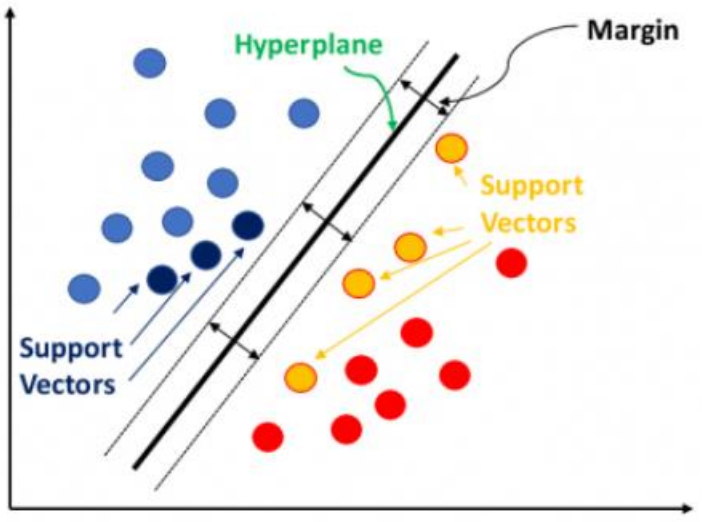
Logistic Regression är en binär klassificeringsmodell vars grund är att prediktera om en instans tillhör en klass eller ej. Modellen använder sig utav Sigmoid funktionen för att beräkna sannolikheten (mellan 0 och 1) att en given bild tillhör en specifik klass (i detta fall siffrorna 0 till 9).



Figur 1. Visuell demonstration på Sigmoid funktionen.

2.2.2 Support Vector Machine

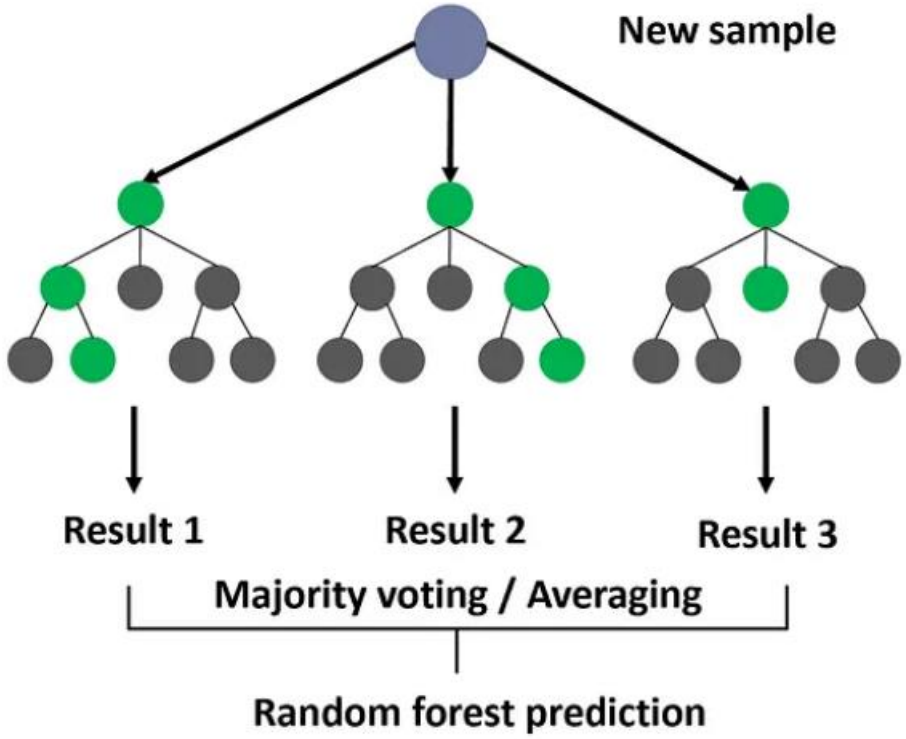
Support Vector Machine är en väldigt flexibel och kraftfull modell som kan användas till flera olika problem. Intuitionen bakom SVMs klassificeringsmodeller är att hitta en så bred väg som möjligt (även kallat hyperparameter) och begränsa margin violations (felklassificeringar). Klassificeringsmodellen vi har använt är Support Vector Clustering med en polynomisk kernel samt med en Gaussian RBF (Radial Basis Function) kernel.



Figur 2. Visuell demonstration på hur SVM fungerar.

2.2.3 Random Forest Classifier

Random Forest Classifier är en ensemble av beslutsträd som använder sig utav majoritetsröstning för att bestämma vilken klass en instans tillhör. Varje enskilt träd gör en prediktion och sedan kombineras resultaten för att bestämma vilken klass objektet tillhör. Denna metod gör att det blir mer precist än t.ex Logistic Regression modellen som enbart har en prediktion.



Figur 3. Visuell demonstration på hur RFC fungerar.

3 Metod

3.1 Dataset

Datasetet som används med denna studie är MNIST (Modified National Institute of Standards and Technology). Detta dataset består av 60 000 träningsbilder och 10 000 testbilder på siffror mellan 0 och 9 som är handskrivna utav studenter och lärare på United States Census Bureau. MNIST författades utav Yann LeCun, Corinna Cortes och Christopher J.C Burges.

Figur 4. Visuell demonstration på MNIST-datasetet.

I denna studie användes 10 000 bilder som delades upp i tränings set bestående av 80% utav datan och ett test-set bestående av 20%. Anledningen till att enbart 10 000 bilder användes var för att snabba upp träningsprocessen av modellerna.

Det är normal praxis inom maskininlärning att dela upp datan i tränings, validerings och testset men beslutet togs att i stället använda cross_val_predict (en funktion i Scikit-learn) som delar upp träningsdatan i fem delar, sedan tränar modellen på fyra utav dessa och validerar modellen på den femte tills varje del har validerats en gång för att få en bra utvärdering på modellerna.

3.2 Explanatory Data Analysis

När det kommer till MNIST-datasetet så är det inte så mycket EDA som behövs göras. Utan datasetet är komplett utan saknade värden och bilderna är standardiserade i 28x28 pixelupplösning för att göra det enkelt att fokusera på själva maskininlärningen.

3.3 Val av modell

För att komma fram till vilken modell som i slutändan ska utvärderas på testdatan har först gridsearchcv använts som är en inbyggd modell i Scikit-learn. Gridsearchcv är en metod som utforskar flera olika parametrar för att hitta den optimala konfigurationen för modellerna.

Efter dom bästa parametrarna har hittats användes cross_val_predict (inbyggd modell i Scikit-learn som använder cross validation) för att träna modellen och sedan göra prediktioner för att kunna se vilken modell som presterar bäst. Resultatet från detta ser vi nedan.

Modell	Precision	Recall	F1-score	Accuracy
LG	0,87	0,87	0,87	0,87
SVM	0,95	0,95	0,95	0,95
RFC	0,95	0,95	0,95	0,95

Tabell1: Resultat för modellerna, för Precision, Recall och F1-score visas genomsnittet.

Som vi ser från tabellen så ger SVM och RFC samma resultat så då får vi utforska vidare med att kolla på hur modellerna presterar på varje enskild siffra.

Classification Report Random Forest:

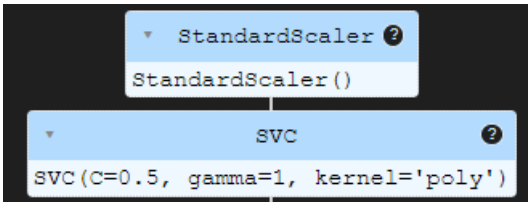
	precision	recall	f1-score	support
0	0.96	0.99	0.98	794
1	0.96	0.98	0.97	911
2	0.94	0.94	0.94	787
3	0.95	0.92	0.93	840
4	0.95	0.94	0.95	769
5	0.96	0.93	0.95	687
6	0.95	0.98	0.96	794
7	0.97	0.95	0.96	854
8	0.95	0.91	0.93	778
9	0.90	0.94	0.92	786
accuracy			0.95	8000
macro avg	0.95	0.95	0.95	8000
weighted avg	0.95	0.95	0.95	8000

Classification Report SVM:

	precision	recall	f1-score	support
0	0.97	0.98	0.98	794
1	0.98	0.98	0.98	911
2	0.93	0.93	0.93	787
3	0.95	0.93	0.94	840
4	0.92	0.97	0.94	769
5	0.96	0.94	0.95	687
6	0.98	0.96	0.97	794
7	0.95	0.94	0.95	854
8	0.91	0.95	0.93	778
9	0.93	0.92	0.93	786
accuracy			0.95	8000
macro avg	0.95	0.95	0.95	8000
weighted avg	0.95	0.95	0.95	8000

Figur 5. Bild på Classification Report för RFC (vänster) och SVM (höger).

Som vi ser ifrån bilden så presterar modellerna i stort sett lika bra. Det som blev avgörande för vilken modells som skulle användas var att RFC hade 90% precision på nior vilket är det lägsta resultatet på alla siffror på alla olika scores någon utav modellerna gör. Därför valdes SVM som slutgiltig modell.



Figur 6. Bild på slutgiltig modellarkitektur.

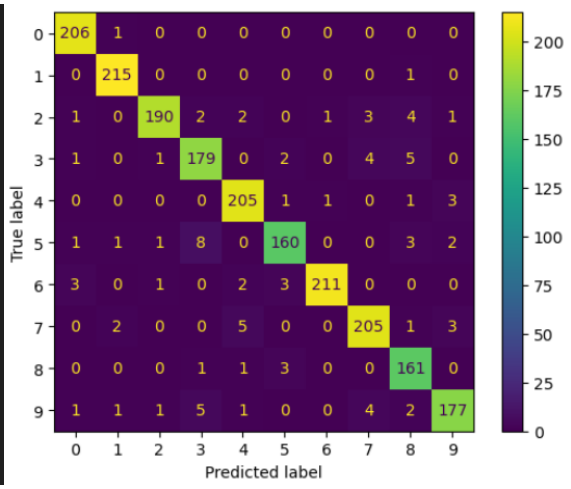
4 Resultat och Diskussion

4.1 Resultat

Resultatet vi fick på vår testdata ser ut som följande:

Classification Report SVM :

	precision	recall	f1-score	support
0	0.97	1.00	0.98	207
1	0.98	1.00	0.99	216
2	0.98	0.93	0.95	204
3	0.92	0.93	0.93	192
4	0.95	0.97	0.96	211
5	0.95	0.91	0.93	176
6	0.99	0.96	0.97	220
7	0.95	0.95	0.95	216
8	0.90	0.97	0.94	166
9	0.95	0.92	0.94	192
accuracy			0.95	2000
macro avg	0.95	0.95	0.95	2000
weighted avg	0.95	0.95	0.95	2000



Figur 7. Bild på slutgiltig Classification Report (vänster) samt Confusion Matrix(höger) från vår SVM modell.

Kollar vi snabbt på rapporten så kan det se ut som att den presterar bättre men snittet är detsamma som på träningsdatan ,vilket är ett väldigt bra resultat. Modellen lyckas prediktera rätt siffra med en accuracy på 95% på tidigare osedd data. Detta betyder att modellen inte överanpassar sig på träningsdatan utan lyckas generalisera bra på osedd data.

4.2 Diskussion

4.2.1 Fråga 1: Hur går maskininlärningsflödet till

Maskininlärningsflödet är egentligen ganska simpelt. Man börjar med att ladda ner sitt data-set, utforskar sin data och städar upp den om det finns null-värden eller om kolumner eller rader behövs tas bort (EDA). Man delar upp sin data i tränings, validerings och testset, eller tränings och testset som vi har gjort. Man väljer vilka modeller man vill utforska och tränar dessa och sedan utvärderar vilken modell som presterar bäst för att sedan använda den på tidigare osedd data.

4.2.2 Fråga 2: Vilken modell presterar bäst på MNIST-datasetet

Av modellerna vi använde oss utav så presterade RFC och SVM lika bra. Utifrån rapporten kan man ej objektivt säga att den ena presterade bättre än den andra. Det man hade kunnat göra är att utforska modellerna ännu mer med fler parametrar för att finputsa dom för ett ännu bättre resultat.

4.2.3 Slutsatser

Detta blir lite som en fortsättning på fråga 2. Den stora slutsatsen är att en simpel modell som Logistic Regression presterar förvånansvärt bra med tanke på hur lite som krävs för att skapa denna modell. Man hade kunnat bygga vidare på denna modell med en one vs rest modell för att få ett ännu bättre resultat som kanske skulle kunna utmana både RFC och SVM.

När det kommer till RFC och SVM så finns det inte mycket mer att säga. Båda modeller är stabila och flexibla som hade kunnat utforskats mer med fler parametrar som tidigare nämnts.

Skulle jag göra om denna studie hade jag använt one vs rest på min Logistic Regression modell och även utforskat fler kernels för min SVM modell samt hur jag hade kunnat göra min RFC modell bättre.

5 Teoretiska frågor

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

Det är praxis inom maskininlärning att dela upp sin data i tränings, validerings och test-set. Man använder träningsdatan till att träna sina olika modeller på olika hyperparametrar, sen utvärderar man sina modeller på valideringsdatan och väljer där modellen som presterar bäst. Man tränar sedan om sin valda modell på tränings och valideringsdatan ihop och sedan använder man testdatan till att utvärdera modellen för att få en överblick på hur den presterar på osedd 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 skulle kunna använda sig utav k-fold cross validation. Med denna funktion kan hon splitta sin träningsdata i n olika delar. Vi säger att Julia splittar sin data i 5 olika delar via k-fold så tränar modellen sig på 4 olika delar och sedan validerar på den femte delen och gör detta då tills varje del har validerats en gång. Sedan kan hon välja modell beroende på vilken som har fått bäst genomsnittlig score.

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

Regressionsproblem handlar om att uppskatta ett kontinuerligt värde från oberoende variabler. Som att uppskatta värdet på ett hus från hur många sovrum, badrum, kvm och utifrån vart huset är lokaliserat.

Några modeller som kan användas för regressionsproblem är linear regression, lasso regression, ridge regression, decision trees och random forest.

4. Hur kan du tolka RMSE och vad används det till: $RMSE$ = √∑(yi − $\hat{y}i$) 2 i

Man kan tolka root mean square error som den genomsnittliga skillnaden mellan uppskattat värde och det faktiska värdet. Inom maskininlärning används RMSE för att utvärdera en modell. Den ger en inblick över hur mycket fel modellen gör med sina prediktioner.

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

Klassificeringsproblem handlar om att identifiera vilken kategori en observation hör hemma. Det kan vara så enkelt som spam eller icke spam. Modeller man kan använda inom klassificeringsproblem är: logistic regression, k-nearest neighbor, decision trees, support vector machines.

Confusion matrix är ett sätt att utvärdera eller se hur en klassificerare presterar. Själva idén bakom denna är att kolla hur många gånger modellen gissat A när B är rätt svar. Om vi kollar på mnist-datasetet så har vi siffror mellan 0 och 9 som modellen ska gissa rätt på. Om vi blandar ihop 4 med 7 kan vi kika på fjärde raden sjunde kolumnen på en confusion matrix för att se detta. Man kan kolla på den på två olika sätt för att se hur bra den fungerar antingen via precision eller recall.

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

K-means modellen är en unsupervised maskininlärningsmodell för att gruppera n observationer i n kluster beroende på längden till klustrets centrum. Denna modell kan tillämpas på kundsegmentering och modellen fungerar bäst när du har en hum om hur många kluster som finns.

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

En maskininlärningsmodell behöver att alla inputs och outputs är numeriska värden. Så Ordinal encoding, one-hot encoding och dummy variable är alla sätt att transformera kategoriska data till numerisk data.

I ordinal encoding så tilldelas varje unikt kategoriskt värde ett heltal. Som exempel har vi olika färger som rankas beroende på vilken ordning dom observeras i datan. Där 1 kan vara röd och 2 kan vara blå osv.

Med One-hot encoding så sätter den värdet 1 på den det är och 0 på de andra. Om vi fortsätter med vårt färgexempel från tidigare så kan röd då vara 1 : 0 : 0, en blå hade varit 0 : 1 : 0 och gul 0 : 0 : 1

Dummy variable är liknande one-hot encoding men gör det simplare. Om vi har 3 kategorier så behöver vi inte 3 olika variabler utan vi kan ha 2 kategorier. 1 det är den färgen och 0 det är ej den färgen. Om vi fortsätter på föregående exempel så om vi vet att 1 : 0 : 0 är röd och 0 : 1 : 0 är blå behöver vi inte 0 : 0 : 1 för gul utan om vi vet att den inte är blå eller röd så vet vi då att den är gul. Så 1: 0 blir röd, 0:1 blir blå och 0:0 blir gul

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?

Kollar man på det svart eller vitt så har båda rätt i sina tolkningar. Jag skulle dock påstå att Julia har mer rätt. Även om det är en nominal variabel så kan det tolkas så att den blir 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 open source framework och kan användas till att skapa dataapplikationer i python för maskininlärning.

6 Självutvärdering

1. Utmaningar:

   Den stora utmaningen jag hade i mitt arbete var med SVM-modellen. Jag har problem att få den att fungera och när jag till slut lyckades så var mina resultat väldigt dåliga. Jag fick börja om från början efter att ha lagt för många timmar på den.

   Jag hade problem med om jag skulle dela upp datan i train, val test eller bara train test split och hur jag skulle utvärdera mina modeller. Jag testade några olika saker så som en egen custom scorer men hittade tillslut cross_val_predict och classification report för att utvärdera mina modeller.

   En annan stor utmaning har varit att skriva denna rapport. Jag har aldrig tidigare skrivit en rapport och detta är nytt vatten för mig. Jag förstod det som om att vi skulle skriva på svenska men hade hellre gjort rapporten på engelska då allt man läser om maskininlärning är just på engelska. Man kan enklare sno citat och förkortningar och även då få in några bra referenser i texten. Men om man nu fick skriva på engelska så blir det nästa gång.

2. Vilket betyg jag förtjänar och varför:

   Jag anser att jag förtjänar ett starkt G. Jag har visat att jag förstår maskininlärningsprocessen och har en förståelse av teorin bakom. Det jag känner som kan dra mig ner är denna rapport, fast jag ser detta som en möjlighet att lära mig skriva rapporter och kan bara förbättras till nästa gång.

3. Något du vill lyfta fram till Antonio?

   Jag tycker att du gör ett väldigt bra jobb! Du är intressant, håller ens motivation uppe på lektioner, skapar bra diskussioner. Fortsätt så!

Källförteckning

Aurélien Géron. (2017). Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent System (2nd edition)

Antonio Prgomet. (n.d) Maskininlärning spellista, Hämtad 20 Mars 2024, från https://www.youtube.com/playlist?list=PLgzaMbMPEHEx9Als3F3sKKXexWnyEKH45

MNIST (n.d) Information om MNIST, hämtad 20 Mars, 2024, från http://yann.lecun.com/exdb/mnist/

Scikit-learn (n.d). Classification_report, hämtad 20 Mars, 2024, från https://scikit-learn.org/stable/modules/generated/sklearn.metrics.classification_report.html

Scikit-learn (n.d). Cross_val_predict, hämtad 20 Mars, 2024, från https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.cross_val_predict.html

Scikit-learn (n.d). GridsearchCV hämtad 20 Mars, 2024, från https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.GridSearchCV.html

Scikit-learn (n.d). Logistic Regression, hämtad 20 Mars, 2024, från https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html

Scikit-learn (n.d). Random Forest Classifier, hämtad 20 Mars, 2024, från https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html

Scikit-learn (n.d). Support Vectore Classification, hämtad 20 Mars, 2024, från: https://scikit-learn.org/stable/modules/generated/sklearn.svm.SVC.html

Scikit-learn (n.d). Support Vector Machine, hämtad 20 Mars, 2024, från: https://scikit-learn.org/stable/modules/svm.html