

Olika modeller för igenkänning av handskrivna siffror

Kunskapskontroll 2 - Maskininlärning



ECUTBILDNING

Anton Grigoriev

EC Utbildning, DS2023

Rapport

202403

Abstract

Supervised and semi-unsupervised models are trained to be able to recognize handwritten digits in images. Online app in Streamlit is created, which imposes certain performance constraints. Because any image can be analyzed, app can be used for fortune telling as in fortune cookie.

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

Innehållsförteckning

Abstract	2
1 Inledning	1
2 Teori.....	2
2.1 Vägledda scikit-learn modeller	2
2.2 Icke-vägledda scikit-learn modeller	2
3 Metod	3
4 Resultat och Diskussion	4
4.1 MNIST och EDA	4
4.2 Vägledda scikit-learn modeller	4
4.2.1 GridsearchCV optimering for ExtraTrees	4
4.3 Streamlit	4
4.4 Icke-vägledda scikit-learn modeller	6
5 Slutsatser	7
6 Teoretiska frågor	8
7 Självutvärdering.....	10
Appendix A	Error! Bookmark not defined.
Källförteckning	11

1 Inledning

Maskininlärning är en viktig del av Data Science. För att få känsla för modellering av komplexa datasets använder jag MNIST (LeCun, 2024) – “database of handwritten digits, size-normalized and centered”. Bilderna föll i tio kategorier (klasser, “lables”): 0 till 9, men vissa varianter är i princip omöjlig att skilja (till exempel 7 som ”lutande etta” och 7 som ”sned sjua”). Andra varianter är bara i dålig handstil, med oslutna ringar (till exempel 4 som ”4” och 4 som ”oavslutad nia”).

Jag kommer att studera hur relativt enkla scikit-learn (Scikit-learn, 2024) väglett Support vektors maskin (SVM) och Beslutsträdmodellerna fungerar med MNIST, kan dem kombineras och vilka är bättre. Sedan kommer jag att skapa Streamlit (Streamlit, 2024) app med den bästa modell för att kunna jobba med vanliga bilder. Jag kommer att jämföra väglettmodeller med (semi)icke- väglett inställning och utvärdera om dem tio klasserna kan igenkännas utan att modell kan se lables.

Syftet med denna rapport är att fullborda krav för Kunskapskontroll 2 i Maskininlärning kurs, för att uppfylla syftet så kommer följande frågeställning(ar) att besvaras:

1. Vilken väglett scikit-learn modell fungerar bra på MNIST och kan den optimeras.
2. Hur används sådan modell i Streamlit app med riktiga bilder.
3. Hur bra jämförs icke- väglett modell med väglett?

2 Teori

Trots att men vissa varianter av handskrivna siffror är i princip omöjlig att skilja, MNIST kan bli modellerad och testdata predikterad med upp till 98.666667% precision med hjälp av neural nätverks modeller.(Gutierrez, 2019)

2.1 Vägledda scikit-learn modeller

Streamlit har två begränsningar: max 1GB minne per app och max 100MB per fil om man använder GitHub på dator eller 25MB per fil om man använder webbinterface. Det finns separat LFS (large file storage) GitHub version. Dem två begränsningar innebär att man inte kan deploya även arkiverade modeller till Streamlit p.g.a. dem kommer överstiga 1GB minnesgräns när de används. På samma anledning är det meningslösa att träna bästa modeller på test + valideringsdata, för att dem blir bara större.

Jag väljer SVM modell som ska vara liten (bara support vektors) och Beslutsträdmodellerna: Extra Trees som ska också vara liten (bara Extra Trees), och Random Forest för att den liknar Extra Trees och kan jämföras med den. P.g.a. överlappning mellan handskrivna former kommer SVM inte lyckas att dela siffror perfekt, se fig. 1. (Géron, 2019)

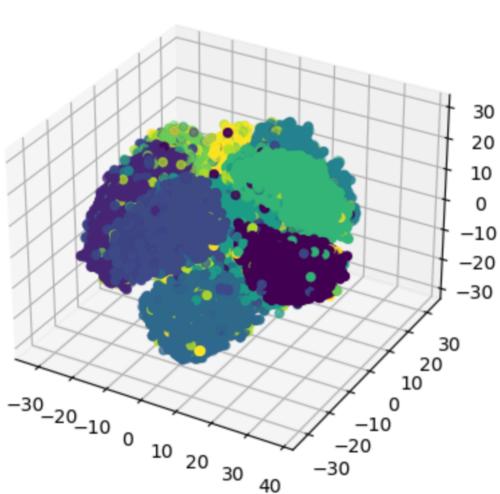
De tre modeller kan kombineras med VotingClassifier.

För parametersoptimering användes GridsearchCV.

2.2 Icke-vägledda scikit-learn modeller

Icke-vägledda scikit-learn modeller är semi-icke-vägledda p.g.a. jag vet att det finns 10 klasser – 10 siffror. Mest avancerad är GaussianMixture som söker efter gaussians i datas distribution.

GaussianMixture kan hjälpas enormt om datadimension reduceras, men man kan tyvärr inte använda reducering för att prediktera nya data.(Maaten, 2019) Istället kommer jag att använda MNIST data separerade i klasser för att studera datakvalitet: vad som kan inte delas i klasser kommer inte delas med t.ex. SVM (men inte tvärtom, p.g.a. SVM är vägledd av klasslabels).



Figur 1: MNIST data 28x28=784d reducerad till 3D med scikit-learn TSNE modell. "Blobbar" av punkter kan modelleras som gaussian-distribuerat 3D klustrar, men de är inte exakt linjärt separerbara med SVM.

3 Metod

Jag använder Anaconda Jupiter Notebook (Anaconda, 2024), där MNIST laddas och gör EDA (Exploratory Data Analysis).

Vägledda modeller tränas, valideras och testas på standard sett, (Géron, 2019) MNIST data är fördelad i 50000:10000:10000 slumpmässiga träning, validering och test delar, med random state = 42.

Större modellers och parametersökning sker övernatt, flera timmar på MacBookPro 2015 med 6 (av 8) CPU i parallell, där "6" är vald i experiment, som visade, att 6 CPU kör lika långt som alla 8 tillsammans p.g.a. overhead belastning.

Modeller sparades med bz2 kompression.

Riktiga bilder av siffror laddades från internet och a) RGBA format konverterades i RGB, sedan RGB i gråskala, och gråa bilder delades i 28x28 fönster och medel färgen i fönstret sparades som pixel.

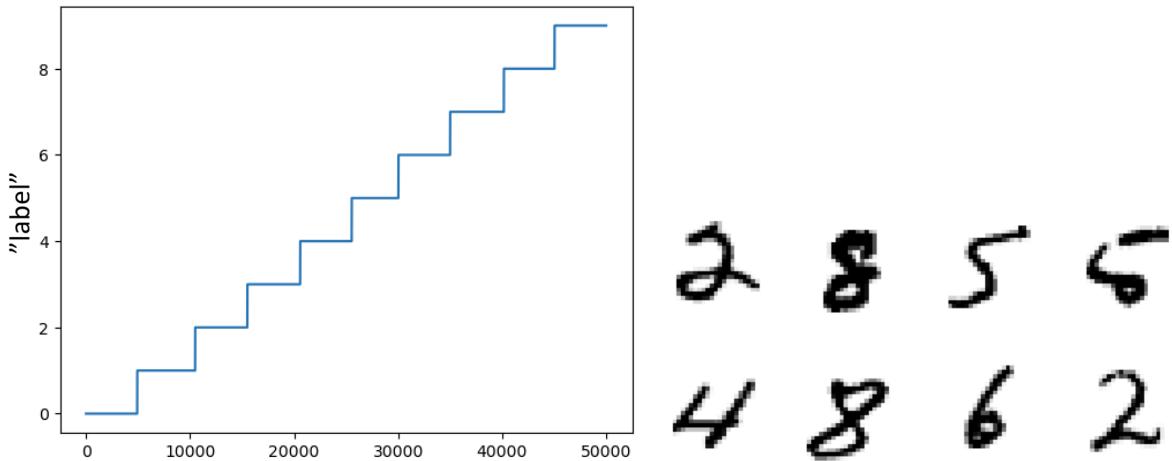
Streamlit modul installerades i anaconda och körs från terminal med /anaconda3/envs/testmojave/bin/streamlit run ML-Kunskapskontroll.py, modell lästes från fil.

ML-Kunskapskontroll.py tillsammans med sparad modell med "git push" fördes till GitHub och deployades till Streamlit.io

Icke-vägledda modeller kördes i Jupiter, datadimension reduceras med TSNE. Klasser matchades till siffror förhand igenom att studera förvirringsmatrisens högsta värde.

4 Resultat och Diskussion

4.1 MNIST och EDA



Figur 2: Vänster:MNIST: sorterade träningslabel visar att alla siffror är representerade. Höger: siffror i MNIST.

Inga konstigheter, väl preparerade data!

4.2 Vägledda scikit-learn modeller

Extra Trees är mycket lika men lite bättre än Random Forest,

Validerings subset accuracy för olika klassificerings modeller	
SVM	0.859%
Extra Trees	0.972%
Random Forest	0.969%
Test subset accuracy för olika klassificerings modeller	
SVM	0.857%
Extra Trees	0.969%
Random Forest	0.965%
Voting (alla tre)	0.965%

Tabell 1: % totalt predikterade rätt med default parametrar.

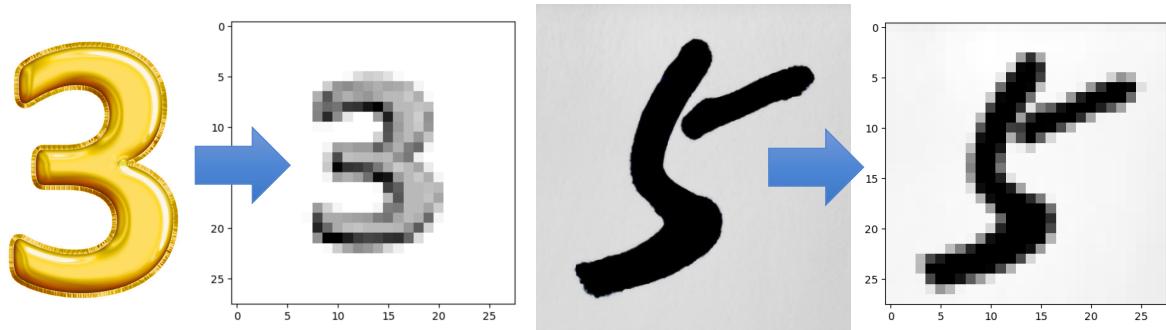
Voting hjälper inte p.g.a. SVM är märkbart värre, och Extra Trees är mycket lika Random Forest.

4.2.1 GridsearchCV optimering for ExtraTrees

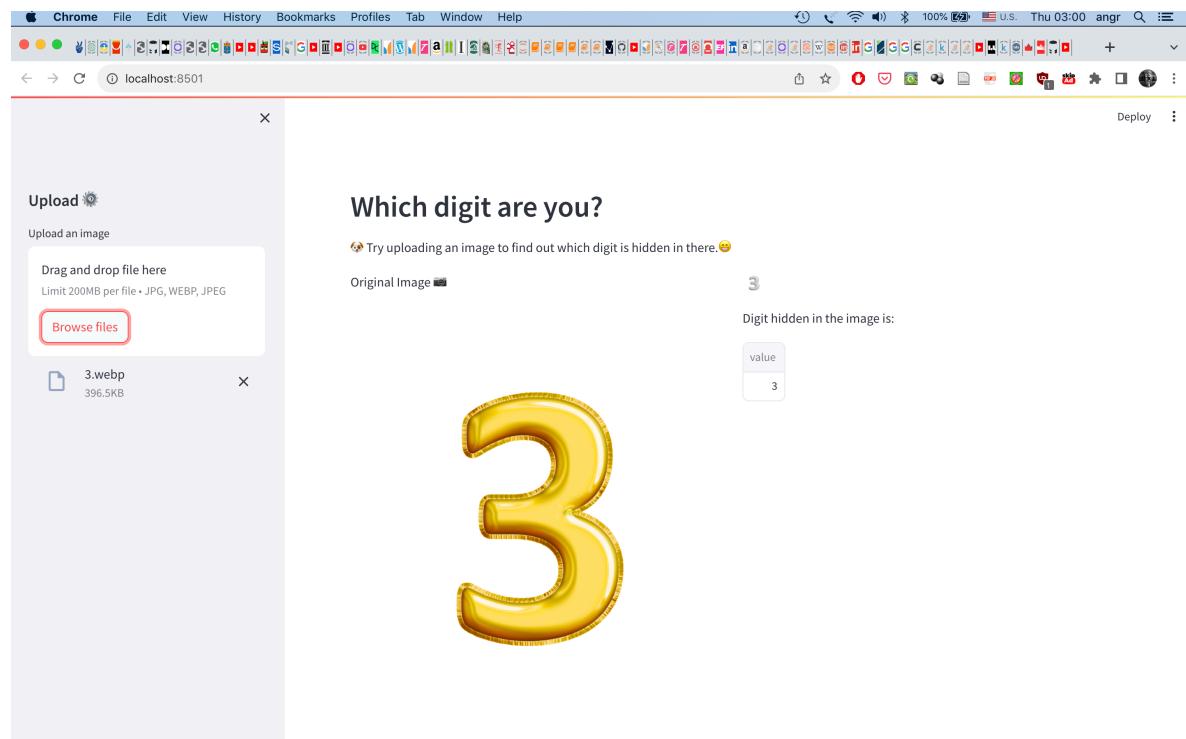
Optimering visade, att man kan höja procent av rätta prediktioner till 97% men bara med kraftigt större modell storlek (n_estimators = 400). Modell beror ganska lite på parametrarna.

4.3 Streamlit

Streamlit app finns här: <https://ml-kunskapskontroll-mkknf2koefsukkh2bflay5.streamlit.app/>

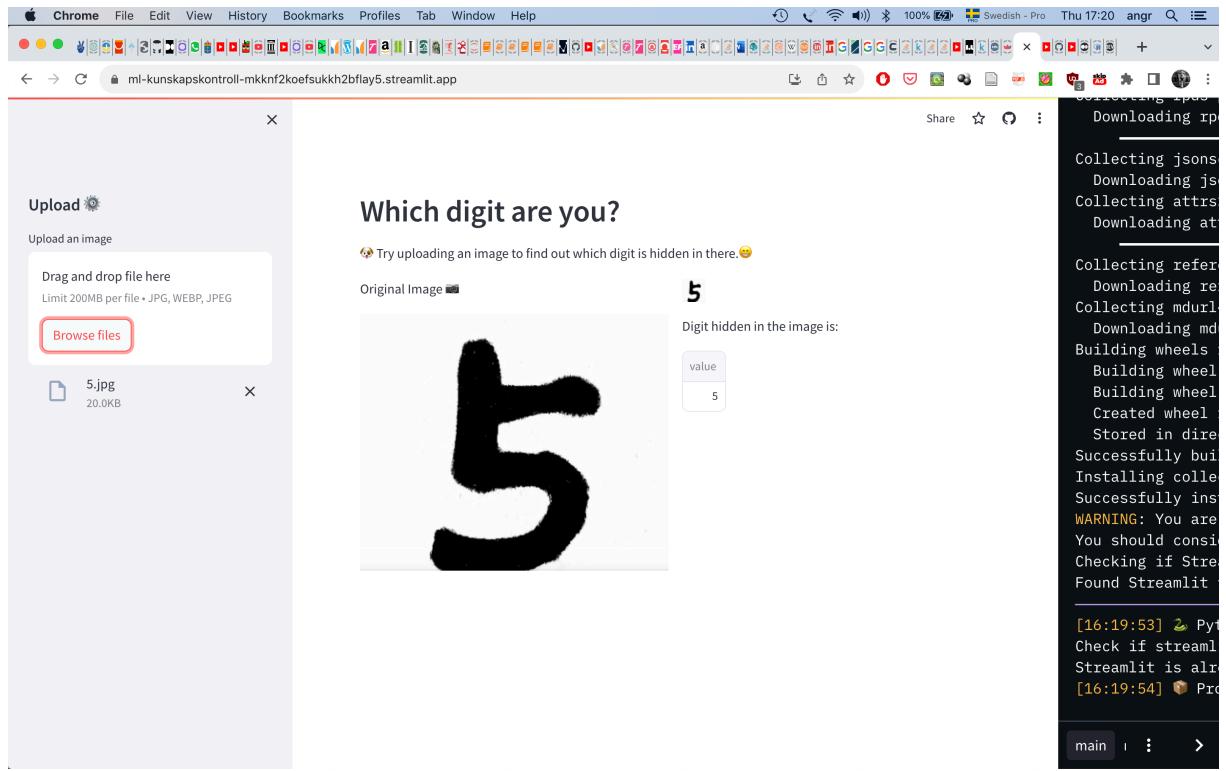


Figur 3: Transformering av bilder.



Figur 4: Streamlit app on localhost.

"Number of estimators" kan inte sättas till 400, och mindre modell med 200 användes.



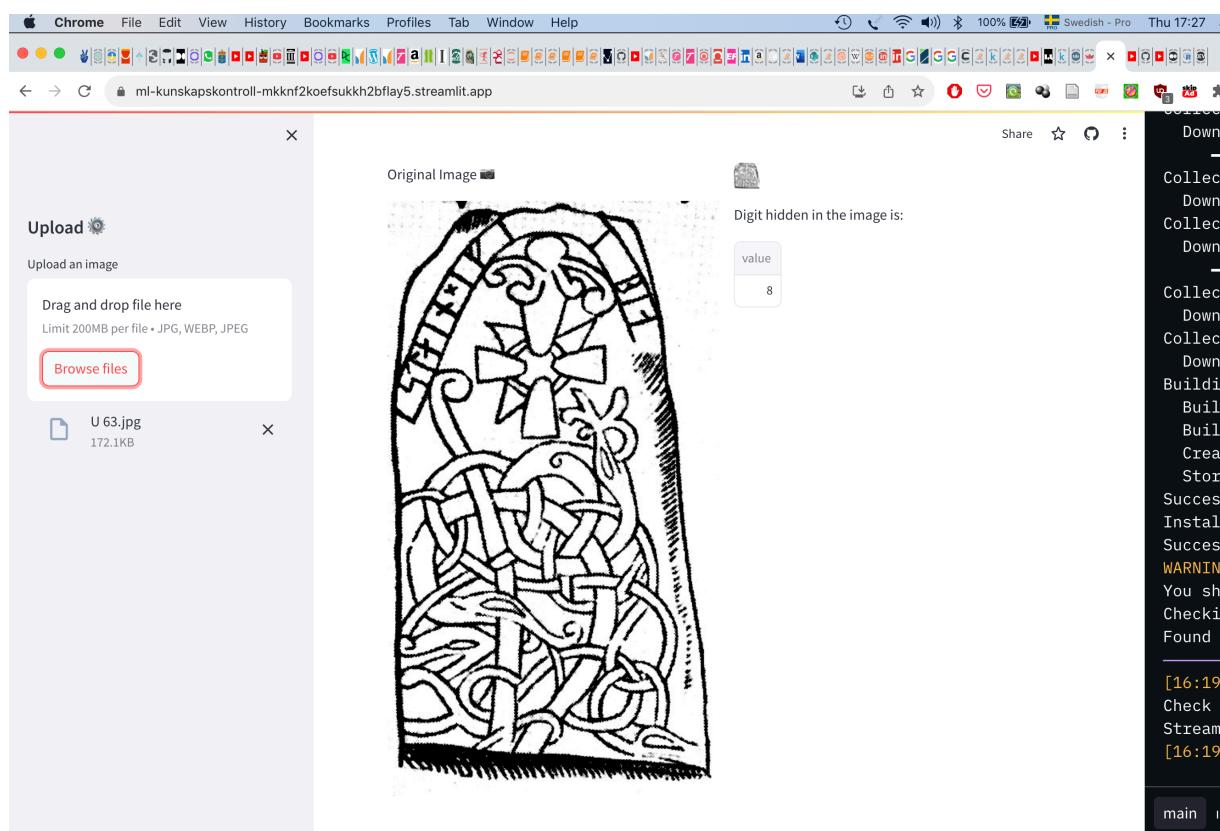
Figur 5: Deployed streamlit app.

4.4 Icke-vägledda scikit-learn modeller

GaussianMixture kan hitta 9 klasser (tionde matchar inte någon siffra). Testdata predikteras med subset accuracy 41%. Med dimensionsreducering (se fig. 1) kan GaussianMixture prediktera 87% siffror.

5 Slutsatser

1. I boken (Géron, 2019) på sidan 105 diskuteras Random Forest felar, och Géron skriver att Extra Trees modellen har bättre bias/varians-avvägning vilket bekräftas av mina beräkningar.
2. Streamlit fungerar med storleks begränsningar. Appen fungerar på vilken bild som helst och kan praktiskt användas som "fortune teller".
3. GaussianMixture kan inte åtskilja ~13% av bilder som jämförs väl med ~15% som inte kan delas md SVM modell. P.g.a. det är bevisat (Maaten, 2019) att klasser kan åtskiljas, förutom 1 till 1.5% som inte kan åtskiljas även med neural networks hjälpen, kan man dra slutsatsen att åtminstone 13 till 15% är så pass dåligt skrivna att även människa kommer att ha svårt att skilja siffror, men i vissa fall troligtvis kommer dator läsa handstil bättre och inte minst fortare.



Figur 6: Streamlit app fungerar på vilken bild som helst.

6 Teoretiska frågor

1. *Kalle delar upp sin data i "Träning", "Validering" och "Test", vad används respektive del för?*
Skapa och träna modeller på tränings, välja modellen på validerings och testa på test. Före testet kan den bästa modellen tränas på tränings + validerings 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 "validerings-dataset"?*
Hon kan använda K-fold cross validering på träningsdatan. Man använder hela träningsdata både som validering och träning, men delar den i K iterationer med 1/K data som används för validering, resten träning och "roterar" valideringsdelen K gånger så att den täcker hela datan. Modell får "betyg" i form av $\sum \text{MSE}/K$.
3. *Vad är "regressionsproblem? Kan du ge några exempel på modeller som används och potentiella tillämpningsområden?*
Maskininlärningsproblem med kontinuerlig beroende variabel Y. T.ex. prediktera pris, probabilitet eller ålder inom finansiell marknad eller sjukvård.
4. *Hur kan du tolka RMSE och vad används det till:*

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2}$$

RMSE är ett viktigt mått för att utvärdera regresionsmodeller, den kan tolkas som distans mellan vektorn \mathbf{y} och vektorn $\mathbf{\hat{y}}$ (den sista med "hatt").

5. *Vad är "klassificeringsproblem? Kan du ge några exempel på modeller som används och potentiella tillämpningsområden? Vad är en "Confusion Matrix"?*
Maskininlärningsproblem med diskret beroende variabel Y. T.ex. olika klasser eller arter, kundbortfall (churn). Varje rad i en förvirringsmatris

representerar en verlig klass, medan varje kolumn representerar en förutsagd klass. Idéen är att räkna antalet gånger instanser av klass A (rad) klassificeras som klass B (kolumn).

6. *Vad är K-means modellen för något? Ge ett exempel på vad det kan tillämpas på.*
Klassificeringsmodellen som är (semi)icke-väglett: man behöver ange antal klasser. Börjar ifrån slumpmässigt gissning och söker stationärt lösning, så att lables och centroids för kluster är stabila. Tillämpas för bildbehandling och semiväglett eller aktivt inlärning (där människor också tittar på data).
7. *Förklara (gärna med ett exempel): Ordinal encoding, one-hot encoding, dummy variable encoding. Se mappen "I8" på GitHub om du behöver repetition.*
Ordinal : varje unikt kategorivärde tilldelas ett heltalsvärde: 3 för "tre"
one-hot : Varje bit representerar en kategori: 0 0 1 för "tre"
dummy : one-hot minus redundancy: 1 0 är "en", 0 1 är "två", 0 0 – "tre".
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.
9. *Kolla följande video om Streamlit:*
<https://www.youtube.com/watch?v=ggDa-RzPP7A&list=PLgzaMbMPEHEx9Als3F3sKKXexWnyEKh45&index=12>
ch besvara följande fråga: - Vad är Streamlit för något och vad kan det användas till?
Service och bibliotek (eller anaconda modul) för att skapa web apps från python skripts. Man kan använda för datamodellering

7 Självutvärdering

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

Jobb-flöde med Streamlit, t.ex. jag kunde enkelt köra app med Anaconda **streamlit run** på dator, men inte den deployed app från GitHub.

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

VG för jag hade uppfyllt kriteria.

3. Något du vill lyfta fram till Antonio?

Introducera bättre jobb-flöde med Streamlit, t.ex. **pipreqs**, skillnaderna mellan hantering av filar och figurer i Jupiter och i Streamlit för att kunna bättre identifiera "bästa praxis" på nätet.

Källförteckning

Anaconda (2024) <https://www.anaconda.com/>

Géron, Aurélien (2019). Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: concepts, tools, and techniques to build intelligent systems (Second edition). O'Reilly Media, Inc. ISBN 978-1-4920-3264-9.

Gutierrez, Antonio Carlos Gomez (2019) <https://www.kaggle.com/code/antoniocgg/mnist-on-scikit-learn-for-beginers>

LeCun, Yann; Corinna Cortes; Christopher J.C. Burges MNIST (2024)
<http://yann.lecun.com/exdb/mnist/>

Maaten, Laurens van der, (2019) https://lvdmaaten.github.io/publications/papers/AISTATS_2009.pdf

Scikit-learn (2024) <https://scikit-learn.org/stable/>

Streamlit, (2024) <https://streamlit.io/>