

# Mini-eksamensprojekt med fokus på kunstig intelligens og maskinelæring

## Introduktion

Følgende opgave er et forslag til et mini-eksamensprojekt for DDU2-holdet. I er dog meget velkomne til at samarbejde på tværs af holdene og I er også meget velkomne til at lave det projekt, som DDU1-holdet skal arbejde med. Se den alternative opgave, som DDU1 får i Lecio.

I skal i de næste uger lave et mini-eksamensprojekt – der vil især være fokus på at I skal lave et (simpelt) produkt og kunne formidle processen og den røde tråd i en skriftlig aflevering. Gruppestørrelse: I skal minimum være 2 og højst 4 i gruppen

## Dokumentation

Den skriftlige dokumentation: \* Skal være i form af en “skabelon” af en opgave, hvori I meget kort beskriver hvad der skal være i de enkelte afsnit og hvor I viser den røde tråd. Derudover skal der være skitser af produktet, der er lavet i forhold til både low og high fidelity. \* Det vigtige er, at I kan vise og dokumentere den røde tråd, i den skriftlige formidling. Hvornår skal hvad komme? Hvor lange skal afsnittene være? Hvad skal de bruges til? Hvordan hænger afsnittene sammen? Hvordan vil I formidle de enkelte afsnit? Hvilke teorier og metoder vil det være relevant at inddrage i de enkelte afsnit? Hvordan skal der laves litteraturhenvisninger og en samlet litteraturliste? Osv.

## Opgavebeskrivelse

I skal som nævnt udforme et simpelt it-produkt der gør brug af kunstig intelligens. Overordnet kan I vælge mellem to typer af produkter (såfremt I ønsker at lave et tredje produkt, så er I meget velkomne til det - men kom lige og snak om det først):

- 1) En streamlit-applikation, der kan bruges til at visualisere et datasæt. Visualisering af store datasæt er særdeles vigtige, da man ofte får mulighed for at se data fra en anden vinkel. Eksempler på sådanne applikationer kan ses her: <https://www.streamlit.io/gallery>. Streamlit har en lang række af forskellige moduler, der kan bruges til at visualisere data:
- 2) En streamlit-applikation, der gør brug af en/flere maskinelæringsmodeller (I kan læse meget mere om de nævnte algortimer på fagets hjemmeside: <https://henriksterner.github.io/IntelligenteSystemer/>):
  - En applikation der gør brug af KNN eller K-means. Eksempelvis til at lave et simpel anbefalingssystem.
  - En applikation der gør brug af en eller flere neurale netværk. Eksempelvis til klassificere billeder eller tekst.

- En applikation der gør brug af lineær regression eller logistisk regression til at forudsige en værdi eller en kategori. Se hjemmesiden for meget mere info om de to algoritmer
  - En applikation der gør brug af naive bayes formel til at klassificere tekst eller billeder. Se hjemmesiden for meget mere info om naive bayes.
  - En applikation der gør brug af en eller flere af de nævnte algoritmer til at løse et problem I selv finder på.
- 3) En Unity-applikation, der gør brug af en eller flere maskinelæringsmodeller. Det kan eksempelvis være at bruge ML-agents til at lave et spil, hvor en eller flere agenter skal lære at spille spillet. I kan også lave et spil, hvor en eller flere agenter skal lære at spille spillet. Der findes en masse dokumentation og tutorials til ML-agents på nettet. Kig eksempelvis på: <https://unity.com/products/machine-learning-agents>
- 4) Webscraping af en side hvor I henter data ned og laver en eller flere maskinelæringsmodeller, der kan bruges til at forudsige noget. Eksempelvis kan I lave en webscraping af en side, der indeholder en masse boligpriser og lave en maskinelæringsmodel, der kan forudsige prisen på en bolig ud fra en række parametre.