

Oppgaver - SOK3023

Maskinlæring for økonomer

Laget av Markus J. Aase

January 29, 2025

Dette dokumentet inneholder oppgaver studentene i SOK-3023 kan jobbe med etter tredje uka av kurset. En del av svarene vil dere kunne finne fra de fysiske forelesningene og/eller kompendiet. Mens noen spørsmål krever at dere oppsøker informasjonen selv i dokumentasjon til Tensorflow eller andre sted på internett.

1. **Kostfunksjon vs. Loss-funksjon**

- (a) Forklar forskjellen mellom en **kostfunksjon** (cost function) og en **loss-funksjon** (loss function).
- (b) Gi eksempler på kostfunksjoner man kan bruke til ulike type maskinlæringsproblemer.

2. **Hensikten med en kostfunksjon**

- (a) Hva er hovedformålet med en kostfunksjon i maskinlæring?
- (b) Hvordan hjelper en godt definert kostfunksjon en modell med å forbedre seg over tid?
- (c) Hvordan kan valg av kostfunksjon påvirke en modells ytelse?

3. **Gradient Descent – Grunnleggende Konsept**

- (a) Forklar hva gradient descent er og hvorfor det brukes i maskinlæring.
- (b) Hva er forskjellen mellom **batch gradient descent**, **stokastisk gradient descent (SGD)** og **mini-batch gradient descent**?

4. **Gradient Descent – Regneoppgave**

Gitt funksjonen:

$$J(w_1, w_2) = w_1^2 + 2w_1w_2 + w_2^2$$

- (a) Finn de partiell deriverte av $J(w_1, w_2)$ med hensyn til w_1 og w_2 .
- (b) Hvis vi starter med $w_1 = 2$ og $w_2 = -1$, og bruker læringsrate $\alpha = 0.1$, hva blir oppdaterte verdier av w_1 og w_2 etter ett steg med gradient descent?

5. Tilbakepropagering (Backpropagation) - for den interesserte

- (a) Hva er tilbakepropagering, og hvorfor er det viktig i nevralt nettverk?
- (b) Hvordan brukes kjerneregelen (chain rule) for å beregne gradientene i et nevralt nettverk?
- (c) Forklar hvordan vektene oppdateres gjennom tilbakepropagering i et fullt tilkoblet nevralt nettverk.

6. Forstå Keras-kode

Nedenfor er en kode for å definere en sekvensiell modell i TensorFlow's Keras API. Beskriv hva hver linje gjør og hvorfor den er viktig.

```
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Flatten, Dropout, Dense
import tensorflow as tf

model = Sequential([
    tf.keras.Input(shape=(28, 28)),
    Flatten(),
    Dense(128, activation='relu'),
    Dropout(0.25),
    Dense(10, activation='softmax')
])
```

- (a) Hva er hensikten med `Flatten()`-laget?
- (b) Hvorfor brukes `Dropout()` i modellen tror du?
- (c) Hva gjør `Dense(10, activation='softmax')` i denne modellen?

7. Endre og tilpasse modellen

- (a) Hvordan ville du modifisert modellen slik at den har **to skjulte lag** med henholdsvis 256 og 128 nevroner?
- (b) Endre aktiveringsfunksjonen i det første skjulte laget fra **relu** til **sigmoid**. Hvordan påvirker dette modellens evne til å lære?
- (c) Hva skjer hvis vi **fjerner dropout-lagene**? Hvordan kan det påvirke modellens ytelse?