2024

Innlevering 1:

DTE-2602: INTRODUKSJON TIL KLASSIFISERING

EIRIK TENNØFJORD – 27/8-24

1 INTRODUKSJON

I denne oppgaven skal vi jobbe med klassifisering av dyr, basert på gitte data. Det er laget et sett med treningsdata hvor selve dyreklassen er kjent. Klassene vi skal kategorisere innenfor er (engelsk):

- Mammal
- Bird
- Reptile
- Fish
- Amphibian
- Bug
- Invertebrate

Videre skal vi klassifisere 76 ukjente dyr til de forskjellige klassene. Ved hjelp av tester som er laget til av lærer i denne innleveringen, så får vi svar på hvor mye av testene som har blitt godkjent når vi kjører koden.

2 TEORI

Det finnes flere forskjellige teorier vi kan benytte for å klassifisere dyrene, men ikke alle metodene eller algoritmene er like effektive. Noen av disse metodene kan også være ressurskrevende, og kreve en god del kunnskap for å kunne gjennomføre/implementere.

Man kan bruke maskinlæring (nevrale nettverk), for å kategorisere disse dyrene. Da kan man benytte treningssettet for å trene om maskinlæringsmodellen. Dette er en metode som kan være noe komplisert sette opp, så tidlig i semesteret.

En annen teori som kan benyttes KNN (K-Nearest Neighbours). Med denne metoden blir den kjente dataen benyttet som en referanse. Når et nytt (ukjent) objekt kommer vil man se på hvilke kjente objekt som er nærmest det nye objektet. Her kan man også bestemme hvor mange kjente punkt man vil sammenligne med. Når man sammenligner med de kjente punktene, vil man se hvilken klasse man har flest naboer av. Det vil da være stor sannsynlighet for at det er denne klassen det ukjente punktet hører til [1, s.189].

3 METODE

For denne oppgaven ble det valgt å benytte seg av metoden KNN. Denne metoden egner seg godt til denne typen oppgave vi har fått. Her får vi benyttet treningsdataen som har en kjent klasse, denne kan vi sammenligne ved hjelp av å sjekke de nærmeste naboene til de ukjente punktene.

For å løse oppgaven ble det for hvert ukjente objekt iterert over hver enkelt av den kjente dataen. Deretter kalkulert en avstand for hver av disse. Dersom avstanden var blant de nærmeste naboene til punktet (basert på antall nærmeste nabo vi benyttet) så ble dette dyret og avstanden lagret i et array.

For antallet nærmeste nabo ble anbefalingen i pensumboken benyttet [1, s. 233]. Siden vi hadde 76 ukjente objekt, brukte vi derfor $\sqrt{76} = 8.7$. Siden dette er et float tall, så ble den bare rundet ned til 8 ved hjelp av int() funksjonen, men man kunne også benyttet 9.

Når hele den kjente listen var iterert gjennom, så ble tallet til den klassen som hadde høyest forekomst i nærmeste nabo listen, returnert i funksjonen.

4 Resultat

Resultatet ble svært bra ifølge den automatiske testingen, og resulterte i 76 av 76 riktige.

```
PS C:\Users\eitenn\Documents\git\ng1_simple_classification> python .\classifier.py
Number of correct classifications: 76 / 76
PS C:\Users\eitenn\Documents\git\ng1_simple_classification> python .\classifier.py
Number of correct classifications: 76 / 76
```

Det ble testet med antallet nærmeste nabo med åtte og med ni. Begge disse resulterte i 76 av 76.

5 Diskusjon

Resultatet var bedre en forventet, ofte kan man ha noen feil i et system som dette. Her var (ifølge automatisk testing) alt riktig.

Test dataen var nok i dette tilfellet fint plassert i forhold til den ferdig klassifiserte dataen, selv om man stiller antallet nærmeste nabo til én, så får man alt riktig. Det er ikke før man kommer til et antall på rundt 60 (som er langt over anbefalt antall) nærmeste naboer, at man begynner å få noen feil.

6 Konklusjon

Problemet ble løst, og vi fikk de resultatene vi ville hat. Koden som er med i innleveringen er nok ikke av den mest effektive, men den er løst ved hjelp av KNN. Det kan også være et bedre alternativ med et NN/ML dersom man har et enda større antall kjent data.

Koden som er i innleveringen, itererer over hvert element i den kjente dataen for hvert element i den ukjente dataen. Dette kan bli svært lite effektivt dersom man har et system av større dimensjoner.

7 Referanser

[1] A.Y. Bhargava, Grokking Algorithms, 1. utg, New York, USA, Manning Publications, 2016