

2a.1)	Hvorfor ønsker vi å dele dataene inn i trening-, validering- og test-sett?
Svar	Vi ønsker å dele dataene inn i trening-, validering- og test-sett for å forsikre at modellen vi utvikler har de evnene vi ønsker at den skal ha. For å oppnå dette er det nødvendig å teste modellen på usett data. Ved å dele opp dataen, forsikrer vi at vi kan evaluere modellen på et subsett av det samme datasettet som modellen ikke har sett før. Dette gjør det mulig å vurdere modellens generaliseringsevne og sikre at den presterer godt også på nye data.

2a.2)	Hvor stor andel av dataene er nå i hver av de tre settene? Ser de tre datasettene ut til å ha lik fordeling for de tre forklaringsvariablene og responsen?
Svar	<ul style="list-style-type: none"> - Opprinnelig sett: 2036 rader - Treningssett: 1221 rader, 59.97% - Valideringssett: 407 rader, 19.99% - Testsett: 408 rader, 20.04% <p>Fordelingen av forklaringsvariablene og responsen i de tre datasettene ser ut til å være god. Det er noen avvik, men disse ligger i utkanten av normalfordelingen. Det er for eksempel litt avvik i 25% blokken, min og max, disse vil igjen påvirke gjennomsnitt og median. Dette er å forvente, da det er et begrenset antall av målinger som havner i utkanten av normalfordelingen. Da disse også må plasseres i et av datasettene, må det forventes litt skjevfordeling som resultat. Responsvariabelen Y er også godt fordelt, dette kan sjekkes ved å finne prosentandelen ikke-suksesser mot suksesser: Treningssett: $693/528 \cdot 100 = 131.25$, Valideringssett: $231/176 \cdot 100 = 131.25$, Testsett: $231/177 \cdot 100 = 130.50$. Vi ser at kvotienten er tilnærmet lik i alle tre datasett, det er et lite avvik i Testsettet, men dette er å forvente som nevnt tidligere.</p>

2a.3)	La oss si at vi hadde valgt League 1 og 2 som treningssett, Championship som valideringssett, og Premier League som testsett. Hvorfor hadde dette vært dumt?
Svar	Å velge forskjellige datasett etter ligaer til trening, testing og validering er ikke lurt da disse datasettene mest sannsynlig har forskjellige karakteristikker, og dermed risikerer vi at modellen ikke generaliserer godt. Modellen vil risikere å lære mønstre som er spesifikke til League 1 og 2, noe som vil gjøre den veldig god til å prediktere kamper fra League 1 og 2, men mest sannsynlig ganske dårlig på Champions League og Premier League kamper. For å forsikre oss om at modellen kan håndtere kamper fra alle ligaene godt, er det bedre å blande datasettene og fordele de slik det er gjort i starten av denne Jupyter-notatboken.

2a.4)	Kommenter kort på hva du ser i plottene og utskriften (maks 5 setninger).
Svar	Vi ser på de empiriske tetthetsplottene at det er nesten fullstendig overlapp på corner_diff og foreelse_diff og noe mindre men fortsatt stor overlapp på skudd_paa_maal_diff, noe som tyder på at corner_diff og foreelse_diff er svært dårlig og skudd_paa_maal er dårlig (men noe bedre enn de to andre) til å klassifisere kampene alene. Vi ser også at det er mye overlapp i kryssplottene, men også her ser vi at det er noe klarere skille i kryssplottene som inkluderer skudd_paa_maal_diff. Dette henter til at hvis vi setter sammen flere av variablene så vil prediksjonen bli bedre. I korrelasjons tabellen kan vi se at det er en moderat

	korrelasjon mellom <code>corner_diff</code> og <code>skudd_paa_maal_diff</code> , noe som gir logisk mening da skudd på mål som blir reddet gjerne resulterer i en korner. Vi ser også at <code>forseelse_diff</code> har svak korrelasjon med alle de andre datapunktene og er dermed en ganske dårlig variabel for klassifikasjonsprosessen.
--	--

2a.5)	Hvilke(n) av de tre variablene tror du vil være god(e) til å bruke til å predikere om det blir hjemmeseier? Begrunn svaret kort (maks 3 setninger).
Svar	Som nevnt i svaret over, ser vi i de empiriske tetthetsplottene at det er litt mindre overlapp i <code>skudd_paa_maal_diff</code> enn det er i de to andre variablene. Jo mindre overlapp det er i et slikt plot jo viktigere er variabelen for treffsikker predikering, dette gir også logisk mening da stor differanse i skudd på mål ofte peker til at det ene laget har mer kontroll på spillet.

2b.1)	I en kamp der <code>skudd_paa_maal_diff</code> er 2, <code>corner_diff</code> er -2 og <code>forseelse_diff</code> er 6, hva er ifølge modellen sannsynligheten for at hjemmelaget vinner? Vis utregninger og/eller kode, og oppgi svaret med tre desimaler.
Svar	Ifølge modellen er sannsynligheten 0.738 for at hjemmelaget vinner. Kode: $\text{sum} = 0.382565 * 2 + (-0.100377 * (2)) + 0.012009 * 6$ $\text{probability} = 1 / (1 + \text{np.exp}(\text{sum}))$

2b.2)	Hvordan kan du tolke verdien av $e^{\beta_{\text{skudd-paa-maal-diff}}}$?
Svar	Verdien av $e^{\beta_{\text{skudd-paa-maal-diff}}}$ tilsier hvor mye oddsen for at y inntreffer multipliseres dersom antall skudd på mål øker med 1. I dette tilfellet multipliseres oddsen med ca. 1.466 per skudd på mål.

2b.3)	Hva angir feilraten til modellen? Hvilket datasett er feilraten regnet ut fra? Er du fornøyd med verdien til feilraten?
Svar	Feilraten i modellen angir den andelen kamper som ble klassifisert i feil kategori ved bruk av prediksjonsmodellen. Programmet regner ut feilraten i valideringsdatasettet. Feilraten er relativt god, da den predikerer riktig nesten 75% av tiden, men det er tydelig rom for forbedring. I statistikk opererer vi ofte med 5% signifikansnivå der vi godtar at vi feilaktig forkaster nullhypotesen kun 5% av tiden, så det er logisk å etterstrebe den samme feilraten her.

2b.4)	Diskuter kort koeffisientene (β – ene) og feilraten endrer seg når <code>forseelse_diff</code> tas ut av modellen (maks 3 setninger).
Svar	Ved å fjerne <code>forseelse_diff</code> fra modellen ser vi en liten økning i koeffisientverdien for <code>corner_diff</code> og en liten nedgang for <code>skudd_paa_maal_diff</code> , med andre ord vil modellen vektlegge cornere litt mer i forhold til antall mål når den predikerer seier. Vi ser også at feilraten gikk en smule ned, men det er ikke en signifikant endring, kun ca. 0.3%

2b.5)	Med den nye modellen: I en kamp der <code>skudd_paa_maal_diff = 2</code> , <code>corner_diff = -2</code> og <code>forseelse_diff = 6</code> , hva er sannsynligheten for at hjemmelaget vinner ifølge den nye modellen? Oppgi svaret med tre desimaler.
Svar	Den estimerte sannsynligheten for seier er akkurat den samme som den gamle modellen: 0.738

2b.6)	Hvis du skal finne en så god som mulig klassifikasjonsmodell med logistisk regresjon, vil du velge modellen med eller uten <code>forseelse_diff</code> som kovariat? Begrunn kort svaret (maks 3 setninger).
Svar	Vi ville nok valgt modellen uten <code>forseelse_diff</code> ettersom feilraten var en smule lavere uten variabelen. Denne forskjellen var ganske minimal, og det hadde derfor vært fordelaktig å se på andre faktorer enn bare feilrate for å avgjøre om <code>forseelse_diff</code> burde inkluderes i modellen eller ikke.

2c.1)	Påstand: kNN kan bare brukes når vi har maksimalt to forklaringsvariabler. Fleip eller fakta?
Svar	Fleip. kNN fungerer helt fint med flere enn to forklaringsvariabler da man kan representere datapunktene som flerdimensjonale vektorer, der hver dimensjon er en forklaringsvariabel. For å finne avstand mellom datapunkter i n dimensjoner kan man bruke Euklids formel for distanse, som fungerer uavhengig av antall forklaringsvariabler/dimensjoner.

2c.2)	Hvilken verdi av k vil du velge?
Svar	Vi ville valgt $k=91$, da dette gir den minste feilraten i valideringssettet. Feilrate: 0.2800982800982801

2d.1)	Gjør logistisk regresjon eller k -nærmeste-nabo-klassifikasjon det best på fotballkampdataene?
Svar	På testsettet får logistisk regresjon og k -nærmeste nabo akkurat den samme feilraten: 0.31862745098039214. k -nærmeste nabo gir litt lavere feilrate enn logistisk regresjon på treningssettet så det mulig at et annet test-data sett ville gitt et annet resultat.

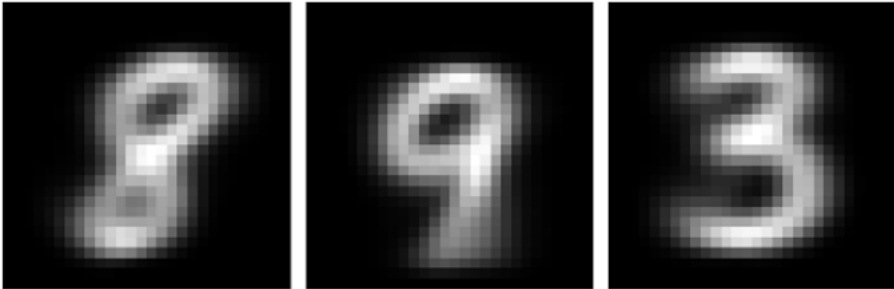
2d.2)	Drøft klassegrensene (plottet under) for de to beste modellene (én logistisk regresjon og én kNN). Hva forteller klassegrensene deg om problemet? Skriv maksimalt 3 setninger.
Svar	Riktigheten av prediksjonene blir dårligere jo nærmere vi kommer midten, og ut ifra den logistiske regresjonen kan vi se at sannsynligheten for å vinne faktisk øker jo lavere corner differansen er. Graden av positiv <code>skudd_paa_maal_diff</code> er naturligvis ganske avgjørende for hvilken klasse datapunktet havner i, og jevne kamper der corner- og skuddifferansen er nær null er det svært mye overlapp, prediksjonen blir derfor sikrere for mer ekstreme diff verdier. Generelt er begge modellene relativt gode for å predikere seier, men valget av variabler fører til veldig høy overlapp når diff verdiene går mot null, hvilket gjør det vanskelig å predikere effektivt.

3a.1)	Hvilke 3 siffer har vi i datasettet? Hvor mange bilder har vi totalt i datasettet?
Svar	De tre sifrene vi har i utvalget av datasettet er 3, 8 og 9. Datasettet består av 6000 bilder.

3a.2)	Hvilket siffer ligner det 500. bildet i datasettet vårt på? Lag et bilde som viser dette sifferet. (Husk at Python begynner nummereringen med 0, og derfor refereres det 500. bildet til [499])
--------------	---

Svar	<p>Det 500. bildet ligner på tallet 9.</p> 
-------------	---

3b.1)	Tegn sentroidene av de 3 klyngene fra K -gjennomsnitt modellen. Tilpass koden over for å plote. Her kan du ta skjermbilde av sentroidene og lime inn i svararket. Hint: Sentroidene har samme format som dataene (de er 384-dimensjonale), og hvis de er representative vil de se ut som tall.
--------------	--

Svar	
-------------	--

3b.2)	Synes du at grupperingen i klynger er relevant og nyttig? Forklar. Maks 3 setninger.
--------------	--

Svar	Klyngene ser ut til å representere de tre sifrene som er i utvalget fra datasettet 3, 8 og 9. Dermed kan de sies å være relevante for vårt utvalg av datasettet.
-------------	--

3b.3)	Vi har valgt $K = 3$ for dette eksempelet fordi vi vil finne klynger som representerer de 3 sifrene. Men generelt er K vilkårlig. Kom opp med et forslag for hvordan man (generelt, ikke nødvendigvis her) best kan velge K . Beskriv i egne ord med maks 3 setninger.
--------------	--

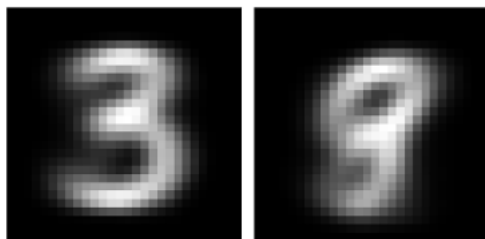
Svar	Artikkelen henvist i oppgaven introduserer to metoder for å finne et forslag for K , «The Elbow Method» og «The Silhouette Method». Elbow-metoden går gjennom mulige verdier for K og finner summen av avstandene mellom punktene og
-------------	--

sentroidene for alle klyngene. Deretter finner man beste verdi for K der summen begynner å bli avtagende.

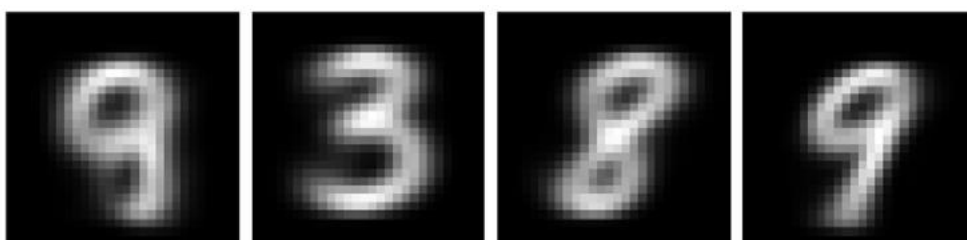
3b.4) Kjør analysen igjen med $K = 2$ og $K = 4$. Synes du de nye grupperingene er relevante?

Svar

$K = 2$:



$K = 4$:

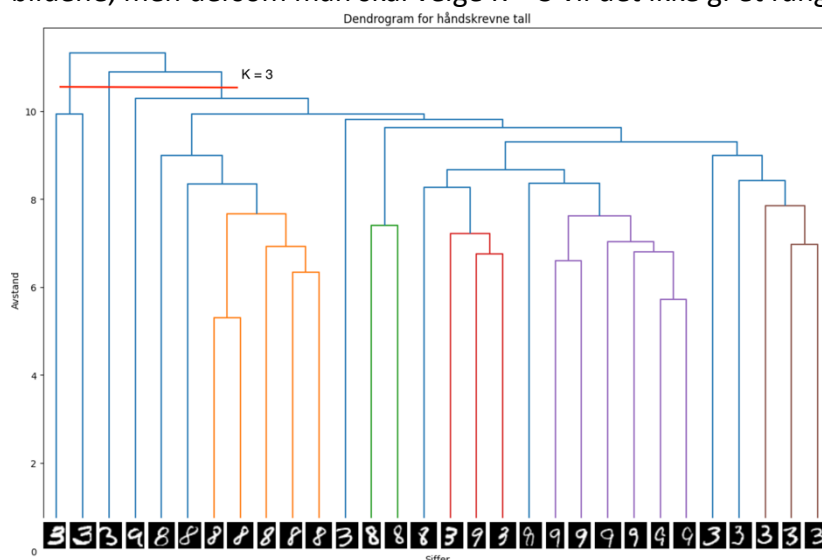


Siden utvalget av datasettet representerer tre distinkte sifre, vil ikke K -verdiene 2 og 4 være optimale. En logisk tilnærming er å lage tre grupperinger som tilsvarer hvert siffer av datasettet. Ved $K = 2$ ser vi at vi får en gruppe som primært inneholder sifferet 3 og annen gruppe som fremstår som en blanding av 8 og 9. Når vi setter $K = 4$ får vi tre grupper som ligner på sifrene i datasettet, men den fjerde gruppen inneholder en blanding av flere sifre.

3c.1) Vurder dendrogrammet nedenfor. Synes du at den hierarkiske grupperingsalgoritmen har laget gode/meningsfulle grupper av bildene? (Maks 3 setninger).

Svar

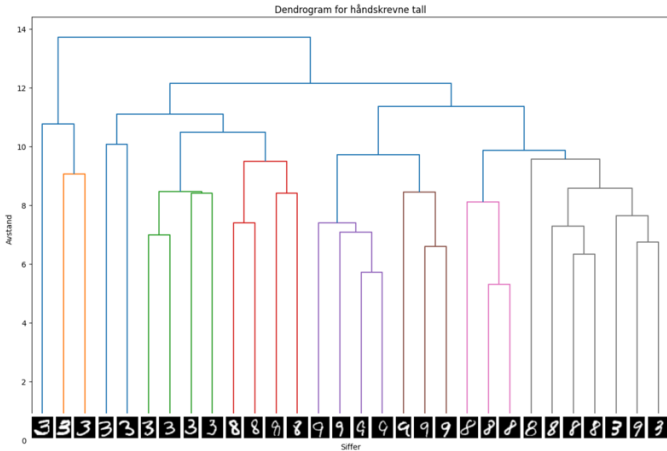
Vi syns grupperingsalgoritmen har laget noen gode/meningsfulle grupper av bildene, men dersom man skal velge $K = 3$ vil det ikke gi et fungerende resultat.



	Man får to grupper som representerer tallet 3, der den ene har kun to tall, den andre kun et. Den siste klyngen inneholder resten av tallene.
--	---

3c.2)	I koden under har vi brukt gjennomsnittskobling (<code>method = 'average'</code>). Hvordan fungerer gjennomsnittskobling? (Maks 3 setninger).
Svar	Med “average” metoden blir gjennomsnittskobling brukt, altså alle MNIST vektorene starter som sin egen klynge, og ved hver iterasjon beregner man en ny klynge basert på gjennomsnittsavstanden mellom alle vektorene sine verdier. Det vil si at for hver iterasjon vil to klynger bli til en. Som vi ser på dendrogramet itereres det gjennom alle vektorene helt til alle vektorene er i en klynge.

3c.3)	Velg en annen metode enn 'average' til å koble klyngene sammen (vi har lært om dette i undervisningen, her heter de <code>single</code> , <code>complete</code> og <code>centriod</code>) og lag et nytt dendrogram ved å tilpasse koden nedenfor. Ser det bedre/verre ut? (Maks 3 setninger).
--------------	---

Svar	<p>Complete:</p>  <p>Vi valgte å bruke “complete”-metoden, og den ser ut til å fungere bedre når det gjelder å koble sammen riktige klynger. Det er verdt å merke seg at metoden ikke klarte å legge alle tallene til riktig klynge, men flere enn det “average”-metoden klarte. Det ser ut til at “complete”-metoden klarte å lage flere mindre og mer kompakte klynger enn “average”-metoden, men når avstanden øker og man forsøker å lage tre separate klynger som inneholder de tre tallene i datasettet, er den ikke helt treffsikker.</p>
-------------	--

3d.1)	Hvis vi skulle brukt en metode for å predikere/klassifisere hvilket siffer et håndskrevet tall er, og ikke bare samle dem i klynge, hva ville du brukt?
Svar	Dette er en typisk oppgave som egner seg godt for nevrale nettverk. Et nevral nettverk bruker noder, vektete kanter og bias for å lagre informasjon i en nodegraf. Deretter benyttes tilbakepropagering for å trene det nevrale nettverket til å gi stadig mer korrekte svar. På denne måten kan man gi et bilde til det nevrale nettverket, som vil omgjøre bildet til en matrise med enkelte pikselverdier. Basert på nettverkets opplæring med andre eksempler fra MNIST-datasettet, kan nettverket gjette hvilket tall matrisen representerer. Dette er da en type

	<p>“supervised learning” fordi vi må gi et svar på hva treningsdataen skal representere for å trene opp modellen til å ta imot ukjent data.</p>
--	---