

Klassificering av handgester från
MYO armband data med hjälp av
maskininlärning och dess
användningsområden för proteser.

Max Tulus

Vår 2023

Datateknik

Åbo Akademi Universitet

Referat

TODO:

- Maybe not refer to this as a study. Consider another way of writing the same content
- Change English terms to Swedish and make sure it has a red thread
- REWRITE AFTER FINISHING DISKUSSION SECTION

Över det senaste åren har tillämpningen av maskininlärning och dess tekniker för proteser, särskilt användningen av elektromyografisignaler (EMG) för att klassificera rörelser och styra protetiska enheter ökat.

Studien undersöker användningen av ett artificiellt neuralt nätverk, särskilt recurrent neural network (RNN), för att bearbeta och klassificera EMG-data. Studien ger en detaljerad metodik som inkluderar förklaring och förberedelse av data, klassificering av data, binär kodning av funktioner och användning av olika neurala nätverks lager som LSTM, dropout, dense och samt en optimerare. Artikeln diskuterar också utvärderingen av neurala nätverk modeller genom confusion matrix, classification rapporter, SVM Classifier och Random Forest Classifier.

Studien visar på potentialen för maskininlärning tekniker att förbättra protetiska enheters funktionalitet genom att klassificera rörelser noggrant och minska belastningen på användarna. Studien lyfter fram fördelarna och nackdelarna med att använda maskininlärning tekniker inom proteser och utforskar den potentiella framtida tillämpningen av denna teknologi. Forskningen drar slutsatsen att maskininlärning tekniker kan förbättra protetiska enheters funktionalitet avsevärt och ge en bättre livskvalitet för protesanvändare.

Innehåll

Referat	1
Innehåll	2
Inledning och Bakgrund (EDIT THIS)	4
Proteser	4
Elektromyografie (EMG)	5
Maskininlärning TODO: Specify more?????	6
Data prosessering	7
Klassificering	7
Kodning	8
Artificial Neural Network (ANN)	9
Recurrent Neural Network (RNN)	10
Övervakat lärande	11
Epochs	12
Kodförklarning	13
Dataförklarning TODO: Simplify and change it more from the original	13
Dataförberedning	14
Klassificering av datan NEEDS TEXT	14
Binär kodning funktioner NEEDS TEXT	15
RNN	15
Lager	15
LSTM (Long Short Term Memory)	16
Dropout	16
Dense	17
Optimizer	17
Epochs	17
Presentering av resultat	17
Resultat Utvärdering	20
Neurala Nätverkets Förklaring	20
Evaluering av Neutrala Nätverket	20
Confusion Matrix	20
Recurrent NN	20
Random Forest Classifier	20
Diskussion	21
Maskininlärning och dess användningar för proteser	21
Evaluering med hjälp av data och resultaten	21

Hur kan denna datan bli använd?	21
För- och Nackdelar med att använda maskininlärning	21
Referenser	22
Artiklar	22
Böcker	22

Inledning och Bakgrund (EDIT THIS)

TODO:

- Read and edit. Take away duplicate things or unnecessary work
- Add citations (Notes in mendely)
- Change to Overleaf??????????????

Proteser är konstgjorda kroppsdelar som används för att ersätta förlorade eller skadade kroppsdelar. De kan vara till stor hjälp för patienter som har förlorat en arm eller ett ben, eller som har en funktionsnedsättning som gör att de behöver hjälpmedel för att kunna röra sig eller utföra dagliga aktiviteter. Historiskt sett har proteser utvecklats under en lång tid, från enkla trä- och metall proteser till dagens avancerade elektroniska proteser som är både lätta och funktionella. Men det finns också utmaningar för protesanvändare, inklusive begränsad rörelseförmåga, smärta och obekvämlighet. Anpassning till protesen kan också vara ett problem, särskilt för nya användare som inte är vana vid att använda en protes.

Med Maskininlärning som ett område som blivit väldigt populärt, diskuterar denna studie om hur maskininlärning kan bli använd för att förbättra kontrollen och anpassningen till en protes för patienter. Tidigare forskning har visat att det finns begränsningar med dagens proteser, inklusive bristande kontroll och anpassning. Men med maskininlärning och användning av klassificerad ytelektromyografi (sEMG) data av handgester, kan en algoritm förutsäga handgesten och därmed förbättra kontrollen för proteser. Detta kan minska problem med anpassning och göra det lättare för patienter att bli bekväma med sin protes och därmed förbättra deras livskvalitet.

Proteser

Proteser är viktiga för personer som har förlorat en kroppsdel eller har en fysisk funktionsproblem. Funktionella krav, designaspekter och anpassning är avgörande delar av att skapa en effektiv och estetiskt tilltalande protes.

Funktionella krav måste beaktas under designprocessen för att säkerställa att protesledet är hållbart, lättviktigt och tillräckligt starkt för att hantera användarens avsedda aktiviteter. Design element som material, form, textur och färg spelar en viktig roll för att skapa en protes som är bekväm, funktionell och estetiskt tilltalande.

Anpassning av proteser är viktigt, eftersom varje individ har unika fysiska egenskaper, livsstilar och preferenser. Anpassning säkerställer att protesen är utformad för att passa perfekt och är skräddarsydd efter individens behov, vilket gör det möjligt för dem att utföra dagliga aktiviteter med lätthet.

Genom att ta hänsyn till funktionella krav, designaspekter och anpassning kan protesanvändare njuta av större komfort, funktionalitet och självständighet.

Elektromyografie (EMG)

Elektromyografi (EMG) är en medicinsk teknik som används för att mäta den elektriska aktiviteten i skelettmuskler. Tekniken har använts för att utveckla proteser, specifikt för att kontrollera rörelsen i protetiska lemmar.

EMG-signaler kan registreras på två sätt: ytelektromyografi (sEMG) och intramuskulär elektromyografi (iEMG). sEMG är en icke-invasiv teknik som registrerar den elektriska aktiviteten i musklerna från hudens yta, medan iEMG innebär att en nålelektrod placeras direkt i muskeln för att registrera elektriska signaler.

Användningen av EMG-signaler för att styra proteser innebär flera utmaningar, såsom muskeltrötthet, rörelseartefakter och hud impedans. Dessutom kan signalerna variera mycket mellan individer och förändras över tid. För att övervinna dessa utmaningar kan maskininlärning algoritmer användas för att förutsäga användarens avsedda rörelse och förbättra noggrannheten i kontrollen av protesen.

Trots dessa utmaningar har EMG-baserad kontroll av proteser flera fördelar. Det tillåter intuitiv och naturlig kontroll av protetiska lemmar, eftersom användaren kan styra protesen med samma muskelgrupper som de skulle med sitt naturliga lem. EMG-baserade proteser kan också anpassas för att passa användarens individuella behov, vilket ger större komfort och funktionalitet.

Maskininlärning **TODO: Specify more?????**

Maskininlärning är ett spännande område inom artificiell intelligens som innebär att lära datorer att lära sig och förbättra från data utan att explicit programmera dem. Det har en bred tillämpning, från självkörande bilar och ansiktsgenkänning till språköversättning och medicinsk diagnos.

Det finns flera typer av maskininlärning, inklusive övervakat lärande, osynligt lärande och förstärkningslärande. I övervakat lärande tränas en maskininlärningsmodell med hjälp av märkt data, medan modellen i förstärknings lärande tränas på omärkt data. Förstärknings Lärande innebär att träna modellen att fatta beslut baserat på belöningar och straff, liknande hur människor lär sig av erfarenhet.

En spännande tillämpning av maskininlärning är dess potential att förbättra proteser. Genom att analysera signaler från användarens muskler med hjälp av tekniker som ytelektromyografi (sEMG) kan maskininlärning algoritmer förutsäga användarens avsiktliga rörelser och styra en protetisk lem därefter. Denna personliga tillvägagångssätt kan leda till mer intuitiv och naturlig kontroll av protesen, vilket förbättrar användarens livskvalitet.

Det finns dock begränsningar och utmaningar med att använda maskininlärning för proteser. En stor utmaning är behovet av stora mängder högkvalitativ data för att träna algoritmerna. Dessutom kan det finnas etiska och integritetsmässiga frågor relaterade till insamling och användning av personlig data för maskininlärning applikationer. Trots dessa utmaningar har maskininlärning

potentialen att revolutionera prostetikområdet och förbättra livet för miljontals människor.

Data processering

Dataprocessering är ett avgörande steg för att förstå de enorma mängder data som genereras inom olika områden. Det innefattar att samla in och rensa data, analysera och visualisera det, förbereda det för maskininlärning och tillämpa olika metoder för att effektivt bearbeta det.

Insamling och rening av data innebär att hämta data från olika källor, ta bort irrelevant eller dublett data och se till att datan är korrekt och komplett. Dataanalys inkluderar användning av statistiska metoder och visualiseringsverktyg för att identifiera mönster och relationer i data.

Förberedelse av data för maskininlärning innebär normalisering, balansering och omvandling av datan för att säkerställa att algoritmerna kan lära sig effektivt. Detta steg är avgörande i maskininlärning, eftersom kvaliteten på indata bestämmer modellens noggrannhet.

Metoder för dataprocessering inkluderar signalbehandling, brusreducering och egenskapsutvinning, bland andra. Dessa tekniker används för att omvandla rådata till en form som är lämplig för analys och maskininlärning.

Sammanfattningsvis är dataprocessering ett viktigt steg för att omvandla rådata till meningsfulla insikter och kunskap. Det kräver noggrann planering och genomförande för att se till att datan är korrekt, komplett och användbar för olika tillämpningar.

Klassificering

Klassificering är ett avgörande koncept inom maskininlärning, där algoritmer används för att identifiera kategorin för en given inmatning. Det innebär att

identifiera mönster och relationer inom data och används inom en mängd olika tillämpningar.

Klassificeringsalgoritmer, såsom stödvektormaskiner och beslutsträd, används för att klassificera data inom olika områden, inklusive bildigenkänning, taligenkänning och naturligt språkbehandling. Dessa algoritmer bygger på statistiska metoder som identifierar mönster i datan och använder dessa mönster för att göra förutsägelser om framtida data.

Det finns dock flera utmaningar med att klassificera data, som överanpassning och underanpassning. Överanpassning uppstår när en modell är för komplex och passar träningsdatan för nära, medan underanpassning uppstår när en modell är för enkel och misslyckas med att fånga de underliggande mönstren i datan.

För att övervinna dessa utmaningar är det viktigt att ha tillräckligt med data för att träna klassificeringsmodeller. Dataförberedsetekniker, såsom normalisering och balansering, kan också användas för att säkerställa att datan är ordentligt förberedd för klassificeringsalgoritmer.

Kodning

Själva kodningen, eller encoding på engelska, är en viktig del av maskininlärning där data konverteras till en lämplig form för algoritmisk bearbetning. Ett exempel på klassificeringskodning är binär kodning, som konverterar data till binära värden, det vill säga 0 och 1.

Kodning används i många olika delar av maskininlärning, till exempel i bildbehandling, taligenkänning och naturligt språkbehandling. I dessa tillämpningar kan kodningens effektivitet och prestanda vara avgörande för att få korrekta resultat.

Att ha rätt kodning är också viktigt för att förbättra prestanda och effektivitet i klassificeringsmodeller. Genom att använda en lämplig kodning kan modellerna lära sig att bättre identifiera mönster och samband i data.

Dock finns det också utmaningar i kodning av data, till exempel när man hanterar stora mängder data eller när data har olika former och strukturer. Att hantera dessa utmaningar kräver expertis inom kodning och maskininlärning, samt en noggrann förståelse för den data som bearbetas.

Artificial Neural Network (ANN)

Artificial Neural Network (ANN) är en spännande typ av maskininlärningsalgoritm som efterliknar strukturen i den mänskliga hjärnan. Nätverket består av sammanlänkade noder som bearbetar indata och genererar förutsägelser, vilket gör det till ett fascinerande område för forskare och utvecklare.

ANN har använts för en mängd olika uppgifter, från bildigenkänning till naturligt språkbehandling och till och med tids serier förutsägelse. Möjligheterna är oändliga. Tänk dig en ANN som identifierar objekt i en bild med precision, översätter text från ett språk till ett annat i realtid eller prognostiserar framtida aktiekurser med otrolig precision.

Dock kan träningsprocessen för ANNs vara tidskrävande och beräkningsintensiv. För att lära av data kräver ANNs stora mängder märkt information, och processen att optimera nätverkets vikter och bias kan vara komplex. Som ett resultat kan specialiserad hårdvara eller distribuerade beräkningsresurser vara nödvändiga för att träna ANNs effektivt.

Trots dessa utmaningar fortsätter ANNs att vara ett kraftfullt verktyg inom området maskininlärning. När teknologin fortsätter att utvecklas kan vi förvänta oss att ANNs blir ännu mer vanliga inom olika tillämpningar, från självkörande bilar till personlig hälso- och sjukvård. Möjligheterna med ANNs är oändliga, och framtiden ser ljus ut för denna otroliga teknologi.

Recurrent Neural Network (RNN)

Recurrent Neural Networks (RNN) är som superhjältar inom maskininlärningens värld. De är specifikt utformade för att bearbeta sekventiell data och möjliggör för maskiner att analysera tidsserie-data eller data med tidsmässig ordning. Till skillnad från andra typer av neurala nätverk har RNN slingor som tillåter information att passera från en sekvenssteget till nästa, vilket skapar en webb av förbindelser som gör det möjligt för dem att förstå strukturen hos sekventiell data.

RNN har funnit en mängd tillämpningar, från naturlig språkbearbetning till taligenkänning och till och med musikkomposition. En av de mest spännande tillämpningarna är språköversättning. RNN kan ta en mening på ett språk som indata och ge ut samma mening på ett annat språk, vilket gör det lättare för människor att kommunicera över språkbarriärer.

Taligenkänning är en annan område där RNN utmärker sig. De kan ta en ljudsignal som indata och ge ut motsvarande text, vilket gör det lättare för personer med hörselnedsättningar att kommunicera. Dessutom används RNN för handskriftigenkänning, där de kan ta en sekvens av streck som indata och ge ut den motsvarande texten.

För att träna en RNN kan olika metoder användas, inklusive övervakad inlärning och oövervakad inlärning. Vid övervakad inlärning tränas nätverket med märkt data, där indata och förväntade utdata tillhandahålls. Vid oövervakad inlärning

tränas nätverket med omärkt data, där det måste lära sig att identifiera mönster i datan själv.

Det finns olika arkitekturer av RNN, såsom enkla RNN, Long Short-Term Memory (LSTM) nätverk och Gated Recurrent Units (GRU). LSTMs och GRUs har blivit alltmer populära på grund av deras förmåga att övervinna problemet med försvinnande gradienter, vilket är ett vanligt problem vid träning av enkla RNN.

Övervakat lärande

Övervakad inlärning är en spännande värld av maskininlärning, där märkt data används för att träna modeller som gör exakta förutsägelser. Algoritmerna som används i övervakad inlärning fungerar genom att identifiera mönster mellan indatafunktioner och motsvarande mållabels.

Bland de mest populära övervakade inlärningsalgoritmerna finns beslutsträd, logistisk regression och stödvektormaskiner. Beslutsträd skapar trädliknande modeller av beslut och deras möjliga utfall. Logistisk regression beräknar sannolikheten för ett särskilt resultat baserat på indatafunktionerna. Stödvektormaskiner är användbara för att hitta den bästa gränsen mellan klasser av data.

Övervakad inlärning har en mängd olika tillämpningar i den moderna världen, inklusive bild- och taligenkänning, naturlig språkbehandling och förutsägande analys. Bildigenkänningsalgoritmer kan tränas för att känna igen ansikten eller specifika objekt i bilder, medan taligenkänningsalgoritmer kan transkribera talade ord till text.

Det avgörande i övervakad inlärning är tillgängligheten av stora mängder märkt data. Ju mer data som är tillgänglig, desto bättre kommer modellen att kunna upptäcka mönster och göra exakta förutsägelser. Detta gör att förvärv och

märkning av data kan vara en tidskrävande och ofta kostsam process, särskilt för mer komplicerade tillämpningar.

Epochs

Epochs är ett begrepp som har stor betydelse inom maskininlärning. Det hänvisar till antalet gånger som hela träningsdatasetet passerar genom modellen under träningen. Epochs är viktiga för att kontrollera varaktigheten och kvaliteten på träningsprocessen, vilket är avgörande för att säkerställa att modellen fungerar optimalt.

Om för få epochs används kan modellen inte identifiera de underliggande mönstren i datat, vilket leder till underanpassning. Med andra ord kan modellen inte generalisera väl till nya data och kan prestera dåligt på testdataset. Om å andra sidan för många epochs används kan modellen bli för specialiserad på träningsdata och kan prestera dåligt på nya data, vilket kallas överanpassning.

Flera faktorer påverkar det optimala antalet epochs för en specifik modell, inklusive modellens komplexitet, kvaliteten och storleken på träningsdatasetet. För att få tillförlitliga och noggranna resultat från modellen är det viktigt att hitta en balans mellan antalet epochs som används under träningen.

Genom att experimentera med antalet epochs kan vi hitta det optimala antalet som säkerställer att modellen presterar bra på nya data samtidigt som vi undviker överanpassning. Därför spelar den korrekta användningen av epochs en stor roll för den övergripande framgången av en maskininlärningsmodell.

Kodförklaring

Dataförklaring **TODO: Simplify and change it more from the original**

Datainsamlingen består av EMG-upptagningar som erhållits från en MYO-armband som har 8 sensorer placerade på huden på höger underarm. Den elektriska aktiviteten som produceras av musklerna under varje sensor mättes, vilket resulterade i 64 kolumner med EMG-data. Varje rad i datasetet har 8 efterföljande läsningar från alla 8 sensorer, följt av en klassificerad kolumn av data. Data insamlades med en frekvens på 200 Hz, vilket resulterar i att varje rad representerar 40 ms av inspelningstiden.

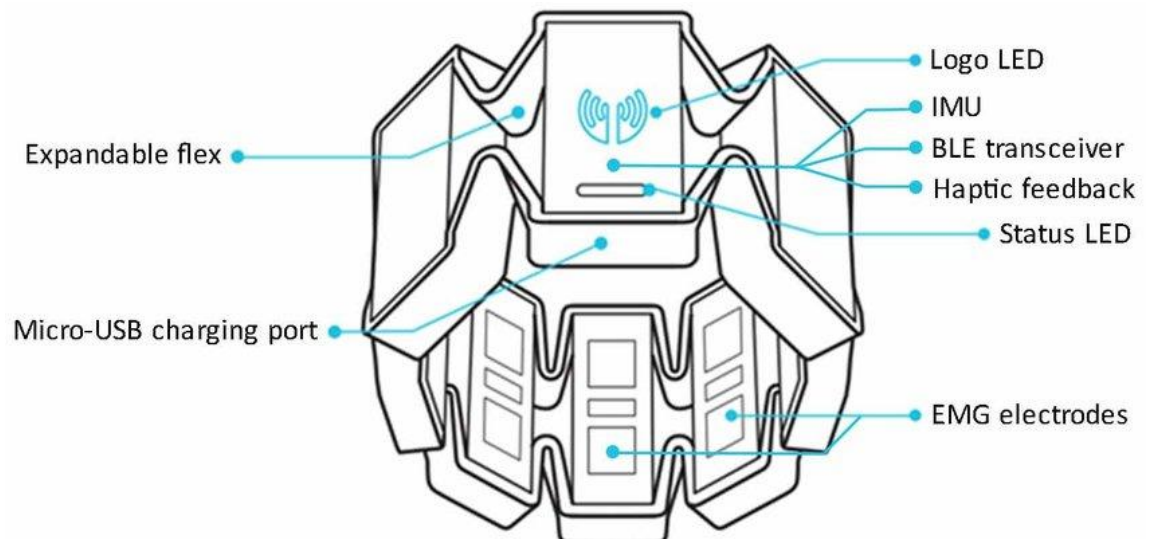


Fig. 1 Diagram på MYO-Armband

https://www.researchgate.net/figure/The-Myo-gesture-control-armband-from-Tha-lmic-Labs-Figure-adapted-from_fig4_323736050 [accessed 26 Mar, 2023]

Varje gest spelades in sex gånger i 20 sekunder, med varje inspelning som startar med att gesten redan var förberedd och hölls, och slutade medan gesten fortfarande hölls. Datainsamlingen är avsedd att användas för gestigenkänning,

där en klassificerare använder de 64 EMG-avläsningarna för att förutsäga motsvarande gestklass.

Dataförberedning

Datainsamlingen genomfördes med hjälp av en app som heter Nukleos som registrerade fyra rörelseklasser - rock, sax, påse och OK - som representeras som 0-3. Alla inspelningar togs från samma högra underarm under en kort tidsperiod, och varje inspelning av en viss gestklass sammanfogades i en .csv-fil med ett motsvarande namn (0-3).

Klassificering av datan **NEEDS TEXT**



Fig. 2 Bild av de klassifierade handgester

<http://www.livelongandtravel.com/5-american-hand-gestures-with-different-meanings-overseas/>

Binär kodning funktioner **NEEDS TEXT**

RNN

Lager

I neurala nätverk är lager byggstenarna som utgör nätverksarkitekturen. Varje lager består av en uppsättning neuroner och dessa neuroner tar emot input från det föregående lagret och genererar utdata som skickas vidare till nästa lager. Det finns många typer av lager, inklusive input-lager, output-lager och dolda lager. Input-lagren tar emot indata och skickar vidare den till nästa lager, medan output-lagren genererar nätverkets slutliga utdata. Dolda lager är däremot ansvariga för att bearbeta och omvandla indata för att generera användbara representationer som kan användas för den slutliga utdatan. Antalet och typen av lager som används i ett nätverk beror på den specifika uppgiften som ska lösas och komplexiteten hos den data som bearbetas.

Model: "sequential"

Layer (type)	Output Shape	Param #
lstm (LSTM)	(None, 8, 50)	11800
dropout (Dropout)	(None, 8, 50)	0
lstm_1 (LSTM)	(None, 8, 50)	20200
dropout_1 (Dropout)	(None, 8, 50)	0
lstm_2 (LSTM)	(None, 8, 50)	20200
dropout_2 (Dropout)	(None, 8, 50)	0
lstm_3 (LSTM)	(None, 50)	20200
dropout_3 (Dropout)	(None, 50)	0
dense (Dense)	(None, 64)	3264
dense_1 (Dense)	(None, 128)	8320
dense_2 (Dense)	(None, 4)	516

=====
Total params: 84,500
Trainable params: 84,500
Non-trainable params: 0
=====

Fig. 3 Tabell av hur neurala nätverkets lager och struktur.

Denna modell är lagat för att analysera sekventiell data med flera funktioner vid varje tidssteg. Den använder en typ av neural nätverk som kallas Long Short-Term Memory (LSTM), som är speciellt utformad för att fånga långsiktiga beroenden i sekventiell data.

Modellen består av 4 LSTM-lager, var och en med 50 enheter, och varje lager returnerar sin utdatasekvens till nästa lager. För att förhindra överanpassning, vilket kan hända när en modell blir för specialiserad på träningsdatan, läggs en dropout-lager till efter varje LSTM-lager med en dropout-hastighet på 20%. Detta innebär att under träning kommer 20% av neuronerna i dropout-lagret slumpmässigt att ignoreras, vilket tvingar de återstående neuronerna att lära sig mer robusta representationer som är mindre beroende av de specifika detaljerna i träningsdatan.

Utgången av det sista LSTM-lagret skickas sedan till ett tätskikt-lager med 64 enheter, följt av ytterligare ett tätskikt-lager med 128 enheter. Det sista utdataskiktet har 4 enheter med en softmax-aktiveringsfunktion, vilket indikerar modellens utdataklasser. Valet av aktiveringsfunktion och utdataskikt beror på den specifika uppgift som modellen används för.

Modellen kompileras med Adam-optimizern, som är en populär optimeringsalgoritm som används inom djupinlärning. Förlustfunktionen som används är kategorisk korsentropi, vilket är vanligt för fler klass klassificeringsproblem, och noggrannhet används som utvärderingsmått.

Övergripande är denna modell utformad för att analysera sekventiell data med en komplex relation mellan flera funktioner vid varje tidssteg, och dess arkitektur består av LSTM-lager, dropout-lager och tätskikt-lager. Användningen av regleringstekniker som dropout och valet av optimeringsalgoritm och förlustfunktion syftar till att förbättra modellens prestanda och generalisering till ny data.

LSTM (Long Short Term Memory)

LSTM (Long Short-Term Memory) är en typ av återkommande neural nätverksarkitektur som kan fånga långvariga beroenden i sekventiella data genom att tillåta information att fortsätta existera över tid. LSTM-nätverk använder en serie grindar för att kontrollera informationsflödet genom nätverket, vilket gör det möjligt att selektivt komma ihåg eller glömma information vid varje tidssteg. Detta gör LSTM särskilt användbara för uppgifter som naturlig språkbehandling, taligenkänning och tidsräkning.

Dropout

Dropout är en regleringsteknik som används i djupinlärning för att förhindra överanpassning genom att slumpmässigt släcka (sätta till noll) vissa neuroner i

en lager under träning. Detta uppmuntrar de kvarvarande neuronerna att lära sig robusta representationer som är mindre beroende av de specifika detaljerna i träningsdatan, vilket kan hjälpa till att förbättra generaliseringsprestanda på nytt, osett data. Dropout har visat sig vara särskilt effektivt för djupa neurala nätverk med många lager, där överanpassning kan vara ett betydande problem.

Dense

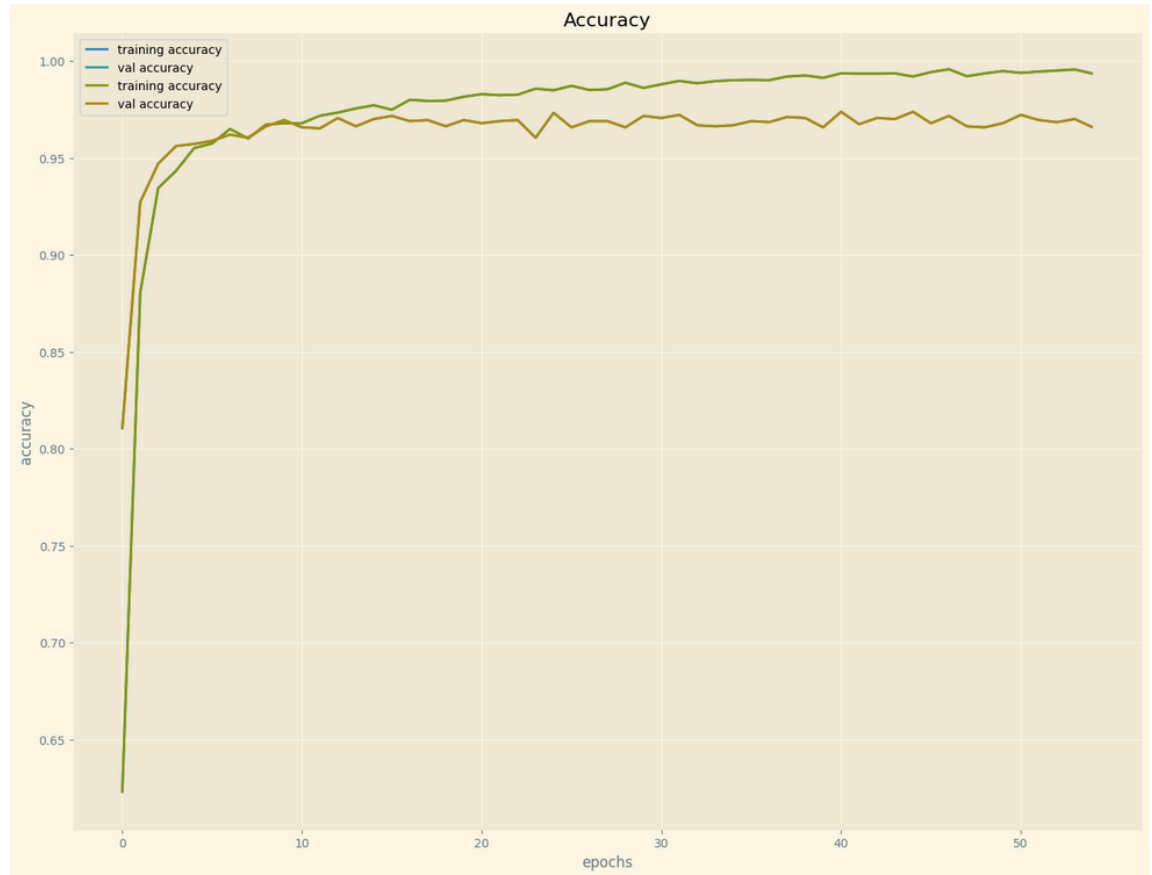
Dense (även kallad fullständigt ansluten) lager är en grundläggande byggsten i djupa neurala nätverk. I ett dense lager är varje neuron i det föregående lagret ansluten till varje neuron i det aktuella lagret, och varje anslutning har en lärande vikt associerad med den. Dense lager används för att kartlägga inmatningsdata till en utgång, och följs vanligtvis av icke-linjära aktiveringsfunktioner för att införa icke-linjäritet i modellen. Flera dense lager kan staplas ovanpå varandra för att bilda ett djupt neuralt nätverk.

Optimizer

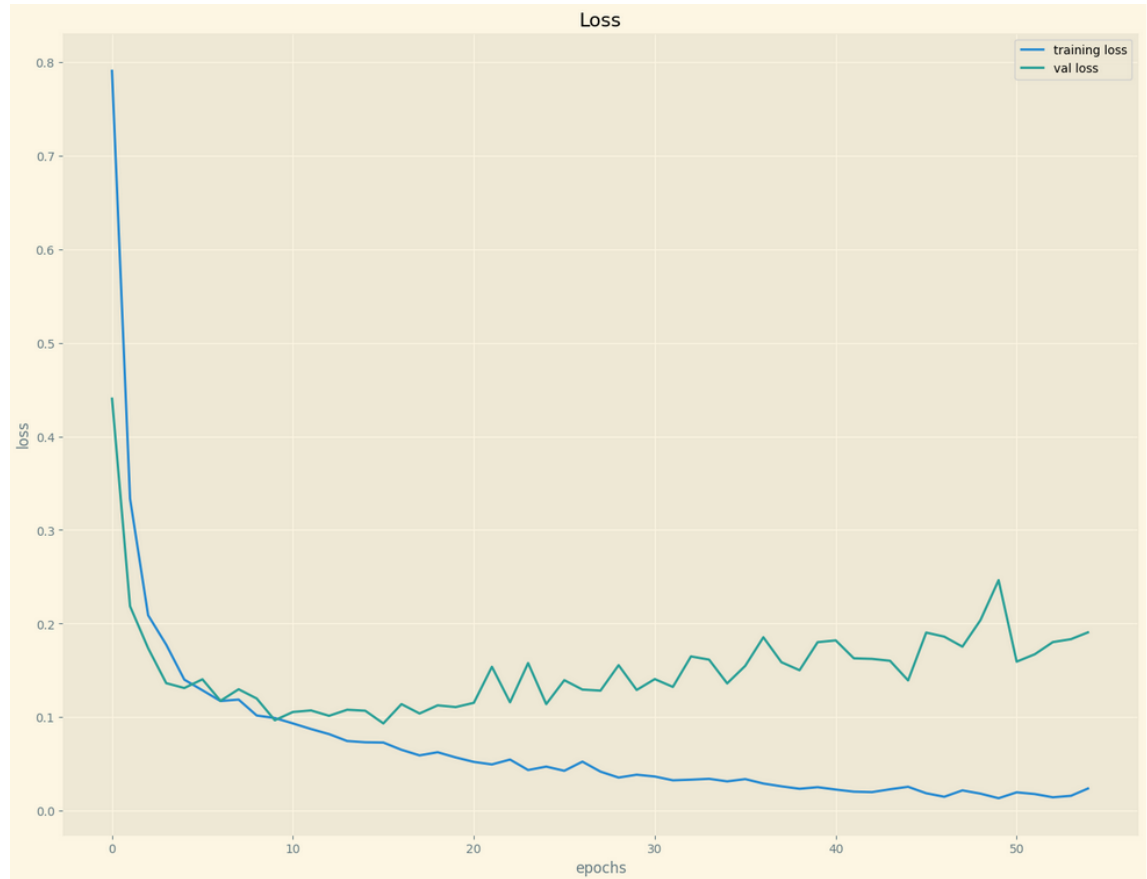
Optimizer är en algoritm som används för att optimera parametrarna (vikter och fördröjningar) i ett neuralt nätverk under träning. Målet med en optimizer är att minimera en förlustfunktion som mäter skillnaden mellan de förutsedda utgångarna från nätverket och de verkliga målen. Optimizers arbetar genom att beräkna gradienterna av förlustfunktionen med avseende på parametrarna och sedan justera parametrarna i riktning mot att minska förlusten. Vanliga optimizers inkluderar stokastisk gradientnedstigning (SGD), Adam och RMSprop, var och en med sina egna avväganden i fråga om hastighet och prestanda.

Epochs

Presentering av resultat



Graf 1: Grafen noggrannhet av modellen. Hur bra den kan förutspå svar basen på tränings data.



Graf 2: Grafen visar hur mycket fel i modellen gissar på träningsdatan

Resultat Utvärdering

Neurala Nätverkets Förklaring

Evaluering av Neurala Nätverket

Classification report - Recurrent NN				
	precision	recall	f1-score	support
0	0.98	1.00	0.99	582
1	0.97	0.98	0.98	581
2	0.99	0.94	0.97	589
3	0.96	0.98	0.97	584
accuracy			0.98	2336
macro avg	0.98	0.98	0.98	2336
weighted avg	0.98	0.98	0.98	2336

Classification report - Random Forest				
	precision	recall	f1-score	support
0	0.81	0.77	0.79	582
1	0.66	0.80	0.72	581
2	0.76	0.71	0.73	589
3	0.63	0.56	0.59	584
accuracy			0.71	2336
macro avg	0.71	0.71	0.71	2336
weighted avg	0.71	0.71	0.71	2336

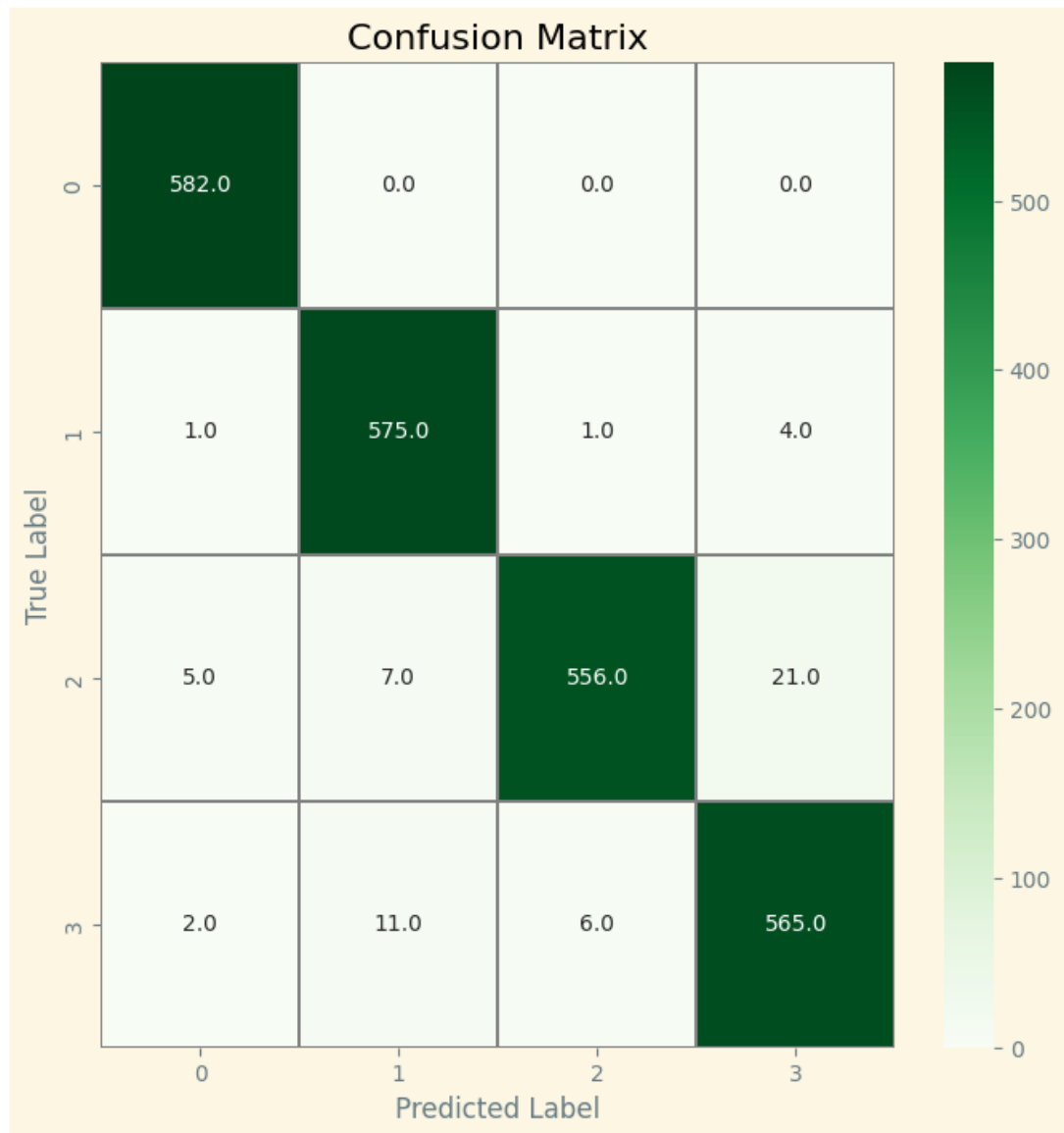
Confusion Matrix

		Predicted	
		Negative (N) -	Positive (P) +
Actual	Negative -	True Negatives (TN)	False Positives (FP) Type I error
	Positive +	False Negatives (FN) Type II error	True Positives (TP)

<https://www.nbshare.io/notebook/626706996/Learn-And-Code-Confusion-Matrix-With-Python/>

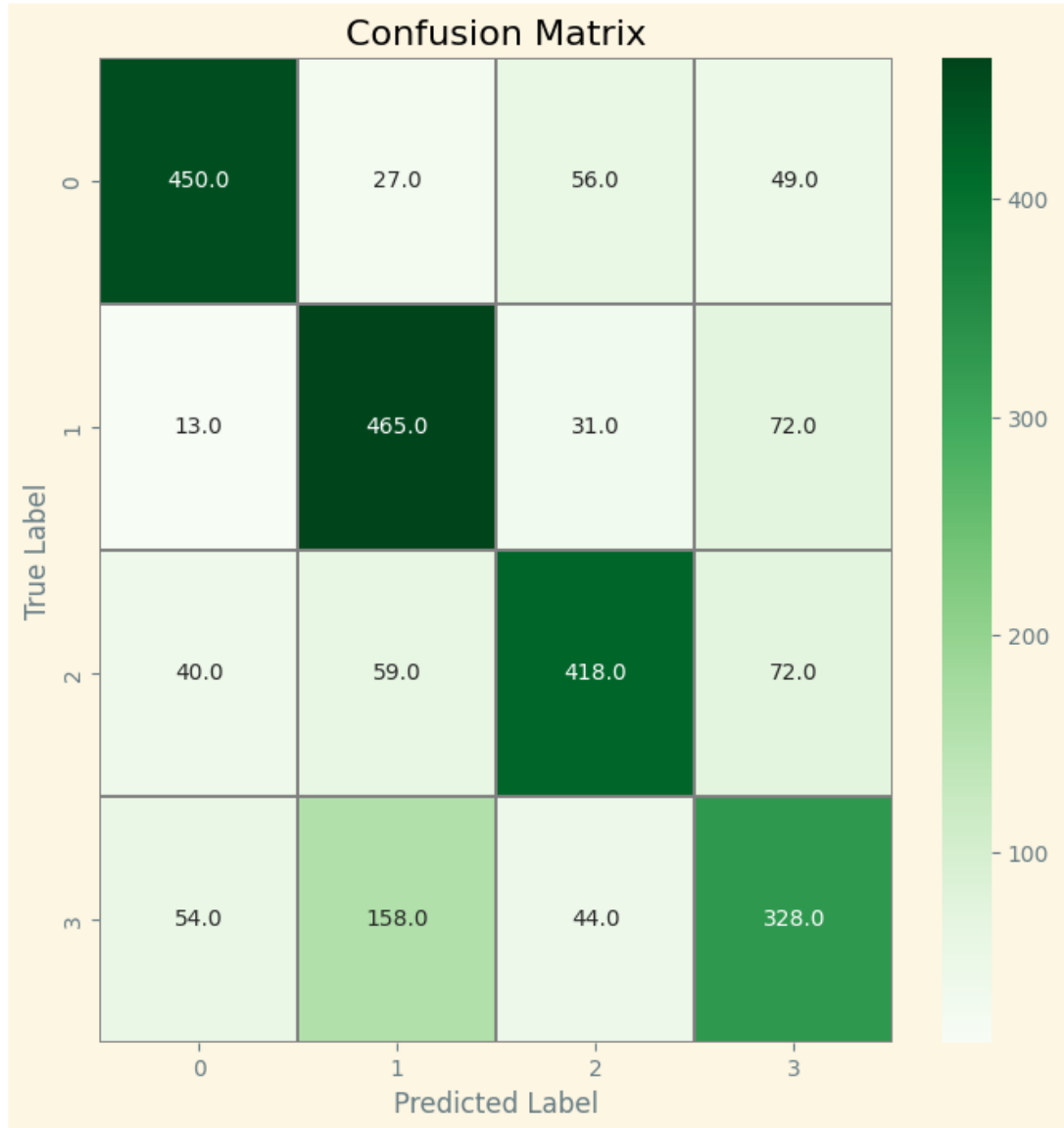
Recurrent NN

Recurrent NN accuracy: 0.9751712328767124



Random Forest Classifier

RandomForestClassifier accuracy: 0.7110445205479452



Diskussion

Maskininlärning och dess användningar för proteser

Evaluering med hjälp av data och resultaten

Hur kan denna datan bli använd?

För- och Nackdelar med att använda maskininlärning

Referenser

Articklar

1. U. Cote-Allard, C. L. Fall, A. Drouin, A. Campeau-Lecours, C. Gosselin, K. Glette, F. Laviolette, and B. Gosselin, "Deep learning for electromyographic hand gesture signal classification using transfer learning," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 27, no. 4, pp. 760–771, 2019.
2. P. Visconti, F. Gaetani, G. A. Zappatore, and P. Primiceri, "Technical features and functionalities of myo armband: An overview on related literature and advanced applications of myoelectric armbands mainly focused on arm prostheses," *International Journal on Smart Sensing and Intelligent Systems*, vol. 11, no. 1, pp. 1–25, 2018.
3. N. Y. Sattar, Z. Kausar, S. A. Usama, U. Farooq, and U. S. Khan, "EMG based control of transhumeral prosthesis using machine learning algorithms," *International Journal of Control, Automation and Systems*, vol. 19, no. 10, pp. 3522–3532, 2021.
4. A. Krasoulis, S. Vijayakumar, and K. Nazarpour, "Multi-grip classification-based prosthesis control with two EMG-IMU sensors," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 28, no. 2, pp. 508–518, 2020.

Böcker

1. L. Massaron and J. P. Mueller, *Machine learning for dummies*. Milano: U. Hoepli, 2019.
2. C. A. Mattmann and S. Penberthy, *Machine learning with tensorflow*. Shelter Island, NY: Manning, 2020.