



SZEGEDI TUDOMÁNYEGYETEM
UNIVERSITAS SCIENTIARUM SZEGEDIENSIS
MÉRNÖKI KAR

***Izomaktivitás feltérképezése EMG érzékelők és mesterséges neurális
hálózatok segítségével***

TDK dolgozat

Béda Zsombor

mechatronikai mérnök MSc

hallgató

Témavezető:

Dr. Sarcevic Péter

főiskolai docens

SZEGED

2024.

TARTALOMJEGYZÉK

BEVEZETÉS	2
1. SZAKIRODALMI ÁTTEKINTÉS.....	3
1.1. Az izomműködés alapjai.....	3
1.1.1. Az elektrofiziológia alapja (Hodgkin – Huxley-Model)	5
1.2. Az EMG alapjai.....	6
1.2.1. Az EMG jelek és jellemzőik	10
1.2.2. Az EMG alkalmazásai és eszközei	12
1.3. Az izomtevékenység felismerésére alkalmazható módszerek.....	16
1.3.1. Jelkezelés és jelfeldolgozás.....	17
1.3.2. Gépi tanulási technikák.....	21
2. ANYAGOK ÉS MÓDSZEREK	27
2.1. Adatgyűjtő rendszer megvalósítása.....	27
2.1.1. EMG szenzorok	27
2.1.2. Mikrovezérlők.....	30
2.1.3. USB leválasztó.....	31
2.1.4. Adatgyűjtő és feldolgozó rendszer felépítése.....	32
2.1.5. Szoftverek	34
2.2. Mérési adatgyűjtés és adatfeldolgozás.....	35
2.2.1. Adatgyűjtési módszerek.....	35
2.2.2. Adatfeldolgozás	38
2.3. Mesterséges neurális hálózat megvalósítása	42
2.3.1. CNN.....	42
2.3.2. KNN (K – Nearest Neighbors).....	44
2.3.3. SVM (Support Vector Machine).....	47
3. EREDMÉNYEK.....	52
3.1. Eredmények.....	52
3.2. Statisztikai elemzés	54
3.3. KÖVETKEZTETÉSEK ÉS JAVASLATOK	57
4. ÖSSZEGZÉS.....	59
5. IRODALOMJEGYZÉK	60
MELLÉKLETEK.....	62

BEVEZETÉS

Manapság nagyon fontos az egészséges, sportos életmód, aminek a megvalósításához egyre több okos eszköz található a piacon egyre több életfunkciót vizsgálva és egyre több adatot biztosítva nekünk a testünk működéséről. A versenyszerűen sportolók számára még fontosabbak a testükre vonatkozó pontos és megfelelő információk, ezért a versenysport világában e célt szolgáló komoly műszerek és vizsgálati módszerek terjedtek el. Ezek a műszerek azért nagyon érdekesek számomra, mert megalkotásukhoz technikai és biológiai tudásra is szükség van. Magam is rendszeresen sportolok, ezért érdeklődöm a diagnosztikai eszközök iránt, így találtam rá az elektromyográfiai mérésekre és eszközökre, amikkel az izmok működését és állapotát tudják vizsgálni, e műszerek az izmok által generált természetes elektromos jeleket mérik és dolgozzák fel.

Tanulmányaim során alaposan megismerkedtem a mesterséges neurális hálókkal, amelyek a természetben előforduló idegi hálózatokat modellezik, és számos összetett számítási feladatra alkalmazhatók, még akkor is, ha a rendszer belső működése nem mindig teljesen átlátható. Ezek a hálózatok különösen hatékonyak a mintafelismerésben és az adatok osztályozásában, ami számos tudományterületen lehetőséget teremt újfajta elemzési módszerek kidolgozására.

A dolgozatom célja mesterséges neurális hálózat alkalmazása felületi elektromyográfiai (EMG) szenzorral mért adatok elemzésére és kategorizálására. Az EMG szenzort használva különböző izomcsoportjaim aktivitását rögzítettem, majd a mért adatokat a mesterséges neurális háló segítségével dolgoztam fel, hogy az egyes izomcsoportok aktivitásmintázatait pontosan megkülönböztethessem.

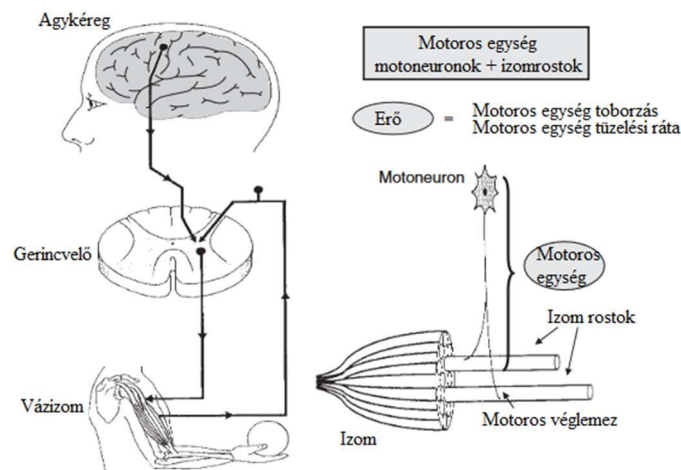
A témához kapcsolódó szakirodalom tanulmányozása során azt tapasztaltam, hogy a hasonló kutatások általában különböző mozgásformák elemzésére fókuszálnak, és ritkábban vizsgálják közvetlenül az egyes izomcsoportok aktivitásait és egymástól való eltérésüket. Úgy gondoltam, hogy érdemes lenne az izomcsoportok szintjén is elvégezni a méréseket és azokat részletesen elemezni, mivel ez új szempontokat adhat az izomműködés és az EMG alapú mintafelismerés megértéséhez.

1. SZAKIRODALMI ÁTTEKINTÉS

1.1. Az izomműködés alapjai

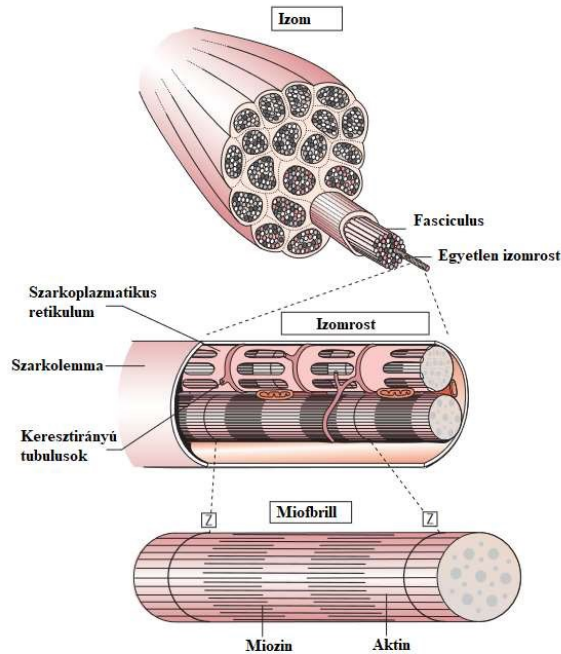
Egy humán mozgatórendszernek belső és külső igények sokféleségével kell megbirkóznia úgy, hogy a saját testünk teljesítési határait figyelembe veszi. A mozgatórendszerünk kihívásai magába foglalja a kifejeteni kívánt erőt és a precíz mozgásokat, a helyes tartást, az összetett mozgásformákat és még az ismétlődő gesztusainkat is. Mivel lehetetlen leírni az összes különböző mozgatórendszer specifikus vezérlési jellemzőit elkülönítve, ezért a mozgásvezérlés alapelveit szükséges körvonalazni, amely kulcsszerepet játszik az erő és a mozgások irányításában az embereknél.

A központi idegrendszer hierarchikus módon szervezett. A „motoros vezérlés” (az emberi izmokban az idegrendszer azon képessége, ami összehangolja és irányítja az izmok mozgását) a premotoros kéregben, a kiegészítő motoros területen és a kéreg egyéb kapcsolódó területein történik. Ezekről a területekről, a kisagyból és bizonyos mértékig a bazális ganglionokból származó bemenetek összpontosulnak a primer motoros kéregben és serkentik vagy gátolják a primer motoros kéreg különböző neuronjait. Ezen kéreg kimenete erős hatással van az agytörzs és a gerincvelő neuronjaira és motoneuronjaira. Létezik egy kapcsolat a corticospinalis pálya és az alfa-motoneuronok között, biztosítva a közvetlen agykérgi ellenőrzést az izomaktivitás felett. Ezt mutatja a következő ábra (1. ábra): (Moritani, Stegeman, & Merletti, 2004)



1. ábra: Alapvető motorvezérlő mechanizmusok sematikus ábrázolása (Moritani, Stegeman, & Merletti, 2004)

A motoros egység fogalma kulcsfontosságú az idegvezetés és az EMG megértéshez. Ez a funkcionális egység egy elülső szarvi idegsejtből, annak axonjából és az összes által beidegzett izomrostból áll. Egészséges állapotban egyetlen motoros egység izomrostjai más motoros egységek izomrostjaival összekapcsolódva helyezkednek el (2. ábra).



2. ábra: Izomszerkezet, amely megmutatja a különböző motoros egységekből származó izomrostok eloszlását. (Michell, 2013)

Egyes motoros egységen belül az összes izomrost vagy I-es vagy II-es típusú, vegyes típusú rostok nincsenek. Az izmorost típusát a beidegződő ideg határozza meg, így a sérülés utáni újrainnerválás (újra beidegződés) után az izomrostok ugyanolyanná válnak, mint az adott motoros egység többi izomrostja, ezáltal kialakul a rosttípus-csoportosulás. Az egyes izmokban az izomrostok különböző típusainak aránya eltérő lehet attól függően, hogy az izom elsősorban a testtartás fentartására szolgál-e (I-es típus dominál), vagy erős, rövid kontrakciókra (II-es típus dominál). Általánosságban a kisebb motoros neuronok vékonyabb axonokkal rendelkeznek, és több I-es típusú izomrostot innerválnak, így az EMG jól vizsgálja őket. Másrészt, a nagyobb motoros neuronok vastag axonokkal rendelkeznek, és általában II-es típusú izomrostokat innerválnak, így nagyobb motoros egységet alkotnak, amelyek kevésbé vizsgálhatók jól EMG-vel. (Michell, 2013)

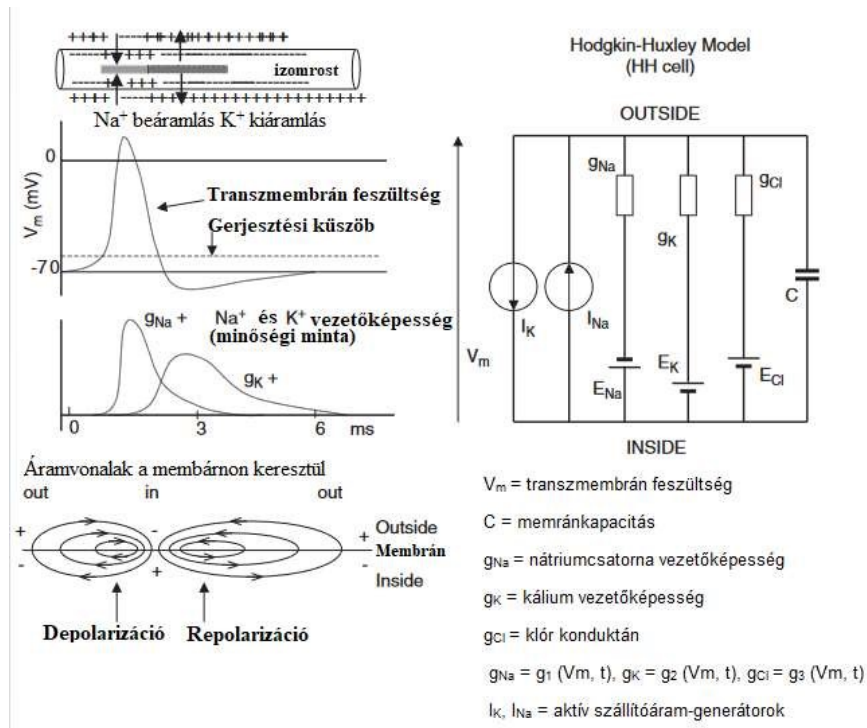
A motoros egység (MU) típusa nemcsak mechanikai és hisztológiai különbségekben nyilvánul meg, hanem az egyedi izomrostok akciós potenciáljában és a motoros egység akciós potenciáljának jellemzőiben is. Wallinga és munkatársai patkány talpmelő izmának (soleus, I-es típus) és az ujjfesztő hosszú izomnak (extensor digitorum longus, EDL, főként II-es típusú) az egyedi izomrostjainak akciós potenciálját vizsgálták. Megállapították, hogy az I-es típusú rostokhoz képest II-es típusú rostok negatívabb nyugalmi potenciállal, nagyobb csúcskitéréssel,

gyorsabb depolarizációs és repolarizációs sebességgel, valamint rövidebb akciós potenciál időtartammal rendelkeznek. (Moritani, Stegeman, & Merletti, 2004)

1.1.1. Az elektrofiziológia alapja (Hodgkin – Huxley-Model)

Az izomrost membránja a bioelektromos jelenségek helye, amelyek az EMG jelet eredményezik. A gerincvelői izomsejtek elektrofiziológiájára vonatkozó legtöbb tanulmány az ingerlékeny sejtekre összpontosított.

Az egyszerűség kedvéért ugyan azt a modellt használják sok tanulmányban az idegsejt membránjának leírására, mint első közelítést az izomsejt membránjára. Ezt a modellt először Hodgkin és Huxley 1952-ben javasolta és egy ekvivalens nemlineáris elektromos áramkört használtak, amely leírja a három fő ioncsatorna viselkedését (3. ábra). A modell kulcsponja a membrán áteresztőképességének dinamikus feszültségfüggő viselkedése a három fő ionra vonatkozóan. A nátrium- és kálium vezetőképességének különböző dinamikus viselkedése miatt akkor jön létre egy átmeneti membránfeszültség-jelenség, amikor egy membránfeszültségi- küszöböt átlépünk. Ezt a jelenséget, amelyet egy „egyszerű” elektronikai áramkör által generált jelenségként lehet elképzelni, akciós potenciálnak nevezik. Ez kémiai változások által indukálható a nátrium vezetőképességében vagy külsőleg alkalmazott elektromos árammal.



3. ábra: Hodgkin-Huxley model és az akciós potenciál generálása (Moritani, Stegeman, & Merletti, 2004)

Összefoglalva a fentieket a mozgásunk folyamata az agy motoros kéregében kezdődik, ahonnan egy neurális jelet küld az agy. Innen a neurális jel a gerincvelőn keresztül közvetítődik az izmoknak motoros neuronok segítségével. A motoros neuronok közvetlenül beidegzik az izmot, ami kalciumionok felszabadulását idézi elő az izomban, és végül mechanikai változást idéz elő. Ez a mechanikai változás magában foglalja a depolarizációt (az elektromechanikus gradiens változását), amelyet azután EMG mérések segítségével érzékelni tudunk. (seedsstudio.com, 2019)

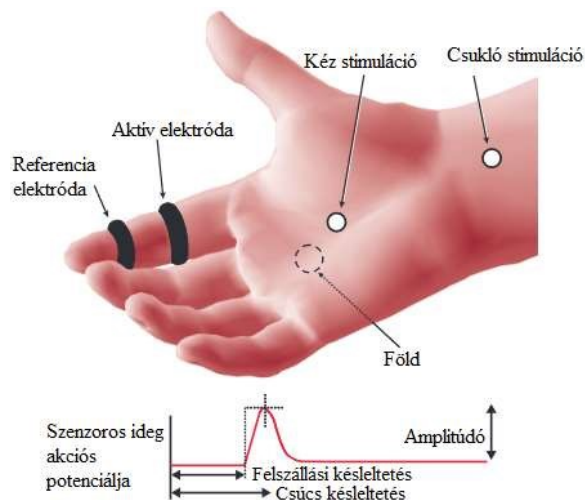
1.2. Az EMG alapjai

Honnan is jön az EMG? Jelentése elektromyográfia, (myo a latinból eredő szó és az izomra utal) ami elsőre ismeretlen lehet, de ez az elektrodiagnostikai tudományokból ered, amit az orvosi szakma alkalmaz neurofiziológiai vizsgálatokhoz. Pontosabban megfogalmazva: „az elektromyográfia egy kísérleti technika, amely a myo-elektromos jelek kialakulásával, rögzítésével és elemzésével foglalkozik. A myo-elektromos jelek az izomrost membránjainak fiziológiai változásából származnak.” (Basmajian & DeLuca: Definition Muscles Alive 2-p.1)

Elektromyográfiai (EMG) vizsgálat során egy izom elektromos aktivitásának mérése és kiértékelése történik, ami egy elektrodiagnostikai orvosi vizsgálat alapvető része. Ez egy nagyon összetett tudomány. Szükség van a vizsgált izmok anatómiájának alapos ismeretére, a használt gép helyes beállításaira és a vizsgálat mögött álló neurofiziológiára. (Jay, Lyn, & Julie, 2016)

Az orvoslás arra használja ezt a tudományt, hogy különböző vizsgálatok segítségével térképezze fel az idegrendszert és az izomműködést. Az „elektrodiagnostikai vizsgálatok” kifejezés valójában számos különböző tesztet foglal magában. A leggyakoribb ezek közül az idegvezetési vizsgálatok és az elektromyográfia (EMG). Gyakran mindkettőt EMG-nek hívják, mert ezek a tesztek az esetek többségében együtt történnek. Ezen tesztek különböző információkkal szolgálnak, ugyanakkor mindkettő értékeli az idegek és/vagy izmok elektromos működését. (Jay, Lyn, & Julie, 2016)

Az NCS-t (Nerve Conduction Studies – idegvezetési tanulmányok) úgy végzik, hogy elektródákat helyeznek a bőrre, és elektromos impulzusokkal stimulálják az idegeket (4. ábra). Ez a szenzoros idegvezetési vizsgálatának alapvető beállítása. A gép nyomkövetést ad a szenzoros ideg akciós potenciáljának állapotáról, ez a SNAP. A motoros idegek tanulmányozásához elektródákat helyeznek egy izom fölé, amely megkapja a beidegzést a tesztelni (stimulálni) kívánt idegből. Az izom elektromos válasza rögzítésre kerül, és ebből meghatározható, hogy milyen gyorsan és hogyan reagál az ideg.



4. ábra: Szenzoros idegvezetési vizsgálat (Peter, 2005)

Az NCS két kategóriába sorolható: motoros és szenzoros idegvezetési vizsgálat. Az autonóm idegrendszer tesztelhető, de klinikailag ritkán van használva. Az NCS bármely hozzáférhető idegen elvégezhető, beleértve a perifériás és agyidegeket. Fontos fogalmak az idegvezetési vizsgálatokhoz:

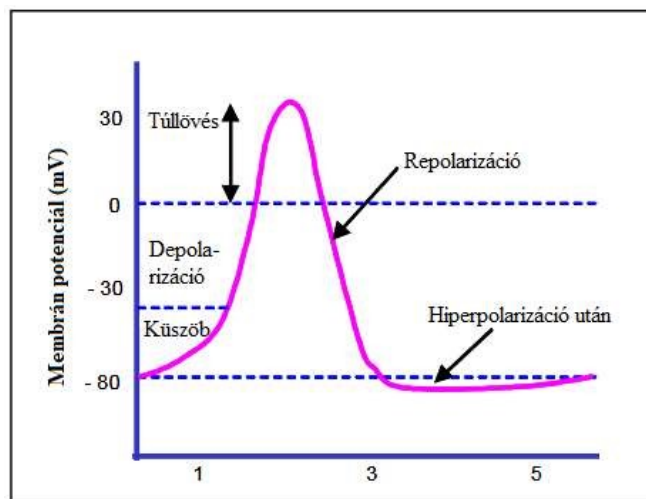
- Akció potenciál – Ez a képernyőn megjelenő látható hullámforma.
- Amplitúdó- Az akció potenciál maximális magassága.
- Vezetési sebesség- Milyen gyorsan halad az impulzus leggyorsabb része (másnéven motorvezetési sebesség vagy szenzoros vezetési sebesség).

Az elektromyográfia (EMG) egy olyan vizsgálat, ami során az izom elektromos aktivitását vizsgálják. Ez az elektromos tevékenység az izomból ered, nincs szükség külső impulzusforrásra. Az EMG azért is különbözik az NCS-től, mert valójában nem foglalja magában az idegek tesztelését. Ugyanakkor közvetetten szolgáltat információt az idegekről (nem szabad elfelejteni, hogy minden izomnak van közvetlen kapcsolata ideggel).

Az izomrostok idegi szabályozáson keresztüli ingerlékenysége az izomfiziológia egyik fő tényezője. Ez a jelenség magyarázható egy félig áteresztő membrán modelljével, amely leírja a sarcolemma elektromos tulajdonságait. Az izomsejt belső és külső tere között ionos egyensúly nyugalmi potenciált képez az izomrost membránján (kb. -80 és 90 mV között, ha nem húzódik össze). Ez a fiziológiai folyamatok által fenntartott potenciálkülönbség (ionpumpa) negatív intracelluláris (sejten belüli) töltést eredményez a külső felülethez képest. Az alfa-motoros elülső szarvsejt aktiválása a gerjesztés vezetését eredményezi a motoros ideg mentén. Miután a transzmitter anyagok felszabadulnak a motoros véglemezekén a motoros egység által

beidegzett izomrostokon végleges potenciál alakul ki. Az izomrost membrán diffúziós jellemzői rövid időre módosulnak, és Na^+ ionok áramlanak be. Ez membrádepolarizációt okoz, amely azonnal helyreáll az ionok visszafelé áramlása végett az aktív ionpumpa mechanizmuson belül.

Ha a Na^+ -on keresztül egy bizonyos küszöbszintet túllépünk beáramlás esetén, a membrán depolarizációja az akciós potenciál gyors változását idézi elő -80 mV -ról egészen $+30 \text{ mV}$ -ra (5. ábra). Ez egy monopoláris elektromos robbanás, amit a repolarizáció szinte azonnal helyreállít, majd a membrán hiperpolarizációs periódusa következik. A motoros véglemezekről indulva az akciós potenciál az izomrost mentén mindkét irányban és az izomrost belsejében, csőrendszeren keresztül terjed.

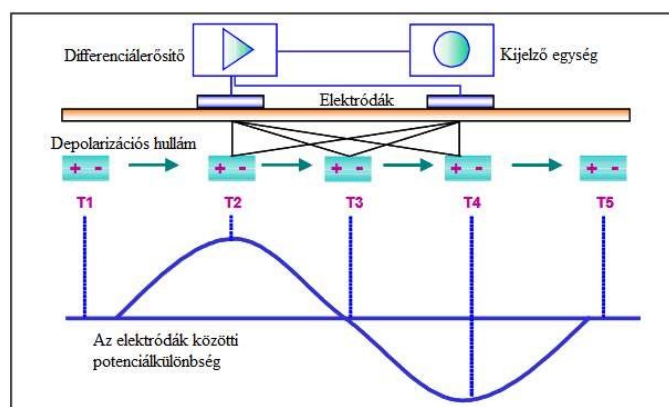


5. ábra: Izomrost akciós potenciál (Peter, 2005)

Ez a gerjesztés kalciumionok felszabadulásához vezet az intercelluláris térben. Az összekapcsolt kémiai folyamatok végül az izomsejt kontraktilis elemeinek lerövidülését eredményezik. Ez a gerjesztést és összehúzódást összekapcsoló modell erősen korrelált kapcsolatot jelent (bár léteznek gyenge gerjesztések, amelyek nem eredményeznek összehúzódást). Gyakorlati szempontból feltételezhető, hogy egy egészséges izomban az izomösszehúzódást bármely formáját a leírt mechanizmusok kísérik. Az EMG-jel az izomrost-membrán akciós potenciálján alapul, amely a fentebb leírt depolarizációs és repolarizációs folyamatokból ered. Ennek a depolarizációs zónának a kiterjedését a szakirodalom körülbelül $1\text{--}3 \text{ mm}^2$ ként írja le. A kezdeti gerjesztést követően ez a zóna $2\text{--}6 \text{ m/s}$ sebességgel halad az izomrost mentén, és áthalad az elektróda oldalán.

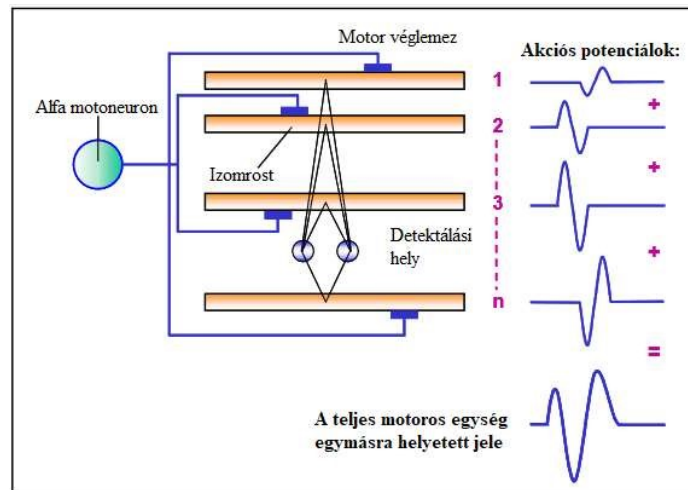
A depolarizációs-repolarizációs ciklus egy elektromos dipólust képez, amely az izomrost felületén halad végig. A kineziológiai EMG mérésekhez tipikusan bipoláris elektróda

konfigurációkat és differenciális erősítést használnak. Az egyszerűség kedvéért első lépésben csak egyetlen izomrost kimutatását szemléltetem. Az 1. és 2. elektródák közötti térbeli távolságoktól függően a dipólus potenciálkülönbséget képez az elektródák között. A látható példában (6. ábra) a T1 időpontban az akciós potenciál generálódik, és az elektródapár felé halad. Az elektródák között növekvő potenciálkülönbség mérhető, amely a T2 pozícióban a legmagasabb. Ha a dipólus egyenlő távolságot ér el az elektródák között, a potenciálkülönbség áthalad a nulla vonalon és a T4 pozícióban lesz a legmagasabb, ami a 2. elektródától való legrövidebb távolságot jelenti.



6. ábra: A „vándorló” elektromos dipólus modellje az izomrostmembránokon (Peter, 2005)

Ez a modell megmagyarázza, hogy a monopoláris akciós potenciál miért hoz létre bipoláris jelet a differenciális erősítési folyamaton belül. Mivel egy motoros egység sok izomrostból áll, az elektródapár „látja” a motoros egységen belüli összes beidegzett rost nagyságát térbeli távolságuktól és felbontásuktól függően. Általában háromfázisú motoregység akciós potenciáljában (MUAP-2) adódik össze, amely a szál geometriai orientációjától és az elektróda helyéhez viszonyított formájától és méretétől függően különbözik (7. ábra). (Peter, 2005)

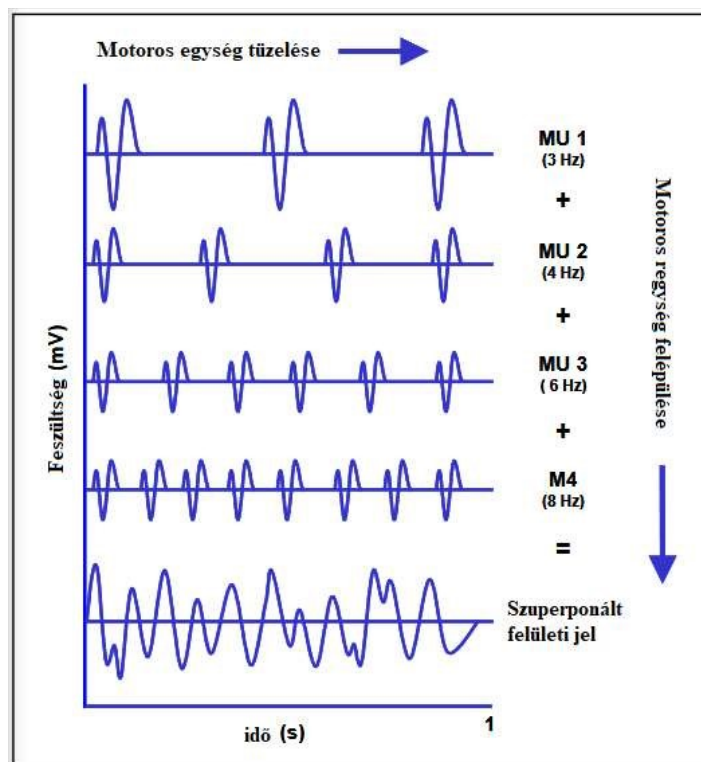


7. ábra: A háromfázisú motoros egység akciós potenciáljának generálása (Peter, 2005)

1.2.1. Az EMG jelek és jellemzőik

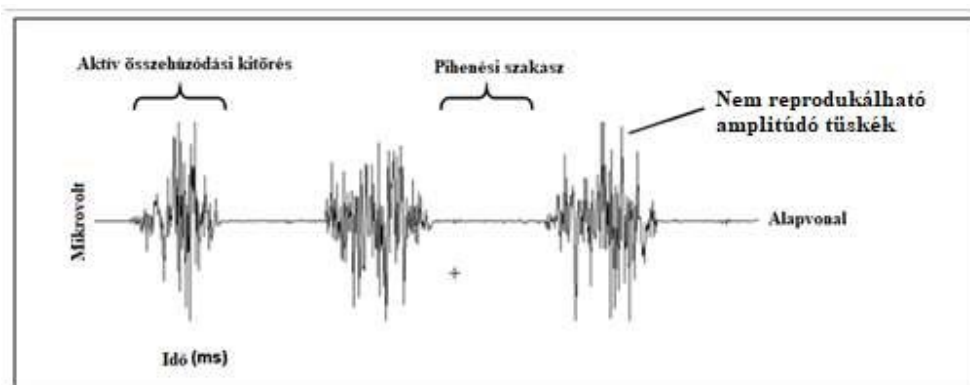
A motoros egység akciós potenciálok szuperpozíciója a kineziológiai vizsgálatok során az elektróda alatt észlelhető összes aktív motoros egység akciós potenciálja elektromosan szuperponálódik, és bipoláris jelként figyelhető meg, amelynek pozitív és negatív amplitúdói szimmetrikusan oszlanak el (átlagértéke nulla). Ezt interferencia mintázatnak nevezik.

A megfigyelt jel nagyságát és sűrűségét befolyásoló két legfontosabb mechanizmus a motoros egység akciós potenciálok felépülése és tüzelési frekvenciája (8. ábra). Ezek a fő irányítási stratégiák a kontrakciós folyamat szabályozására és az érintett izom által kifejtett erő modulálására. Egyszerűen fogalmazva, az EMG jel közvetlenül tükrözi a mért izomban észlelt motoros egységek felépülési és tüzelési jellemzőit. Mivel az emberi kötőszövet és a bőrrétegek aluláteresztő szűrő hatást gyakorolnak az eredeti jelre, a felszíni EMG által elemzett tüzelési frekvencia nem mutatja az eredeti tüzelési és amplitúdó jellemzőket.



8. ábra: A motoros egység aktiválódása és tüzelési frekvenciája modulálja a kifejtett erőt, és ez megjelenik az összegzett EMG jelben. (Peter, 2005)

Egy szűretlen és feldolgozatlan jelet, amely a szuperponált motoros egység akciós potenciálokat (MUAP) észleli, nyers EMG jelnek nevezzük. Az alábbi példában (9. ábra) egy nyers felszíni EMG felvétel (sEMG – surface EMG) látható a bicepsz három statikus összehúzódásáról. Amikor az izom nyugalmi állapotban van, nagyjából zajmentes EMG alapvonal (baseline) figyelhető meg. A nyers jel alapvonal zajszintje számos tényezőtől függ, különösen az EMG erősítő minőségétől, a környezeti zajtól és a mérési feltételek minőségétől. Feltételezve, hogy az erősítő csúcsteljesítményű és a bőr megfelelően elő van készítve, az átlagos alapvonal zajszintje tipikusan nem haladja meg a 3-5 mikrovolt értéket, az ideális cél 1-2 mikrovolt. Az EMG baseline minőségének vizsgálata minden mérés fontos ellenőrző pontja. Ügyelni kell arra, hogy ne értelmezzük a zajokat vagy a mérési problémákat „megnövekedett” alaptevékenységnek vagy izom(hyper)tónusnak.



9. ábra: A biceps három összehúzódásának nyers EMG felvétele (Peter, 2005)

Az egészséges, ellazult izom nem mutat jelentős EMG aktivitást, mivel nincs depolarizáció és akciós potenciál. A nyers EMG jelek természetüknél fogva véletlenszerű alakúak, ami azt jelenti, hogy egy nyers felvétel nem reprodukálható pontosan ugyanabban az alakban. Ez annak köszönhető, hogy az éppen aktivált motoros egységek összetétele folyamatosan változik az elérhető motoros egységek mátrixában/átmérőjében. Ha véletlenül két vagy több motoros egység egyszerre aktiválódik, és közel helyezkednek el az elektródához, erős szuperpozíciós csúcsot hoznak létre. Sima algoritmus (pl. mozgó átlag) alkalmazásával vagy megfelelő amplitúdó paraméter (pl. az egyenesített görbe alatti terület) kiválasztásával a jel nem reprodukálható tartalma kiküszöbölhető vagy legalábbis minimalizálható. A nyers sEMG tartománya +/- 5000 mikrovolt között lehet (sportolóknál) és a frekvenciatartalom általában 6 és 500 Hz között mozog, a legtöbb frekvenciaerősség pedig 20 és 150 Hz között található. (Peter, 2005)

1.2.2. Az EMG alkalmazásai és eszközei

Az emberi test összetett természetes elektromos mechanizmusokkal rendelkezik, amelyek az agytól, a szívtől, az izmoktól, idegrendszerrel és a csontnövekedési mechanizmusoktól származnak. A szervezetünk elektromos rendszerének nagyon fontos szerepe van orvosi területen. Az elektromyográfiai műszerek a vázizomzat által termelt elektromos aktivitás rögzítésére szolgálnak, magukba foglalva a szükséges áramkört a tiszta és erős jelek biztosításához. (Miftachul, 2019)

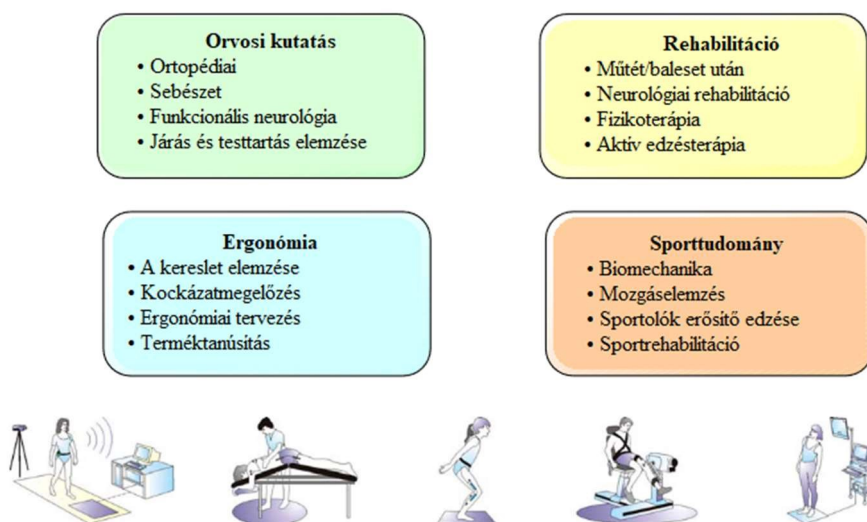
A fiziológiai és biomechanikai alapú kutatások mellett a kineziológiai EMG-t használnak értékelési eszközként az alkalmazott kutatásokhoz, úgy mint: fizioterápia/rehabilitáció, sportedzés és az emberi test interakciói az ipari termékekhez és munkakörülményekhez (10. ábra).

Az orvosi kutatásban az EMG alapvető eszköz az izom- és idegrendszeri működés vizsgálatára. Az ortopédiai kutatásokban segít a csontrendszeri betegségek és sérülések diagnosztizálását. Sebészeti eljárások során az ideg- és izomfunkciók monitorozása révén növeli a beavatkozások pontosságát és biztonságát. Továbbá járás és testtartás elemzés során is segítheti a motoros zavarok diagnosztizálását és kezelését.

Az EMG kiemelt szerepet játszik a rehabilitációban, különösen a műtétek és sérülések utáni felépülésben, az izomfunkciók és idegrendszeri sérülések monitorozásában. A fizioterápiában az EMG adatok segítik a terápiás gyakorlatok optimalizálását, míg modern protézisek vezérlésében lehetővé teszik a természetesebb, intuitív mozgásokat. Az EMG szenzorok az izmok elektromos aktivitását detektálják, és a jeleket jelfeldolgozó algoritmusok mozgásparancsokká alakítják, ami a finommotoros tevékenységek során is pontos irányítást biztosít. A gépi tanulás és neurális hálózatok integrációja tovább javítja a rendszer adaptációját és hosszú távú használhatóságát.

Az ergonómiában az izmok terhelésének mérése érdekében használják, különböző munkakörnyezetekben és feladatok során. Ez segít az igények elemzésében és a munkahelyi kockázatok megelőzésében, valamint munkaállomások és eszközök ergonomikus tervezésében. A szenzor adatokat a termékek és eszközök ergonómiai hatékonyságának tesztelésére és tanúsítására is felhasználják, biztosítva a felhasználók kényelmét.

A sporttudományban az EMG a sportolók teljesítményének javítására és a sérülések megelőzésére szolgál. A biomechanikai elemzések során az izomaktivitás és mozgásminták részletes vizsgálata lehetővé teszi a sportolói technikák optimalizálását. A mozgásanalízis segít az edzésprogramok beállításában, míg az erősítő edzések során az izomaktivitás monitorozása személyre szabott edzések kidolgozását teszi lehetővé. A sportrehabilitációban az EMG az izomaktivitás nyomon követésével segíti a sérült sportolók hatékony felépülését.

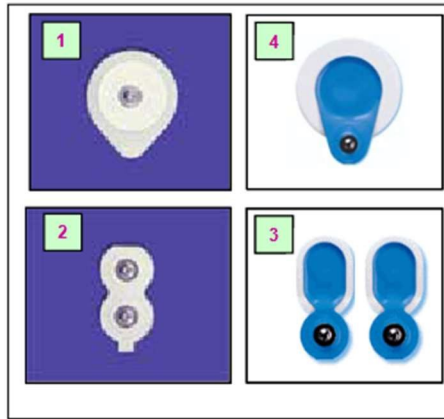


10. ábra: Az EMG széles körű használata (Peter, 2005)

Az általam használt eljárás a felületi elektródás EMG (sEMG) Non-invazív jellegük miatt a legtöbb esetben felületi elektródákat használnak kineziológiai vizsgálatoknál. A könnyű kezelhetőség előnye mellett fő korlátjuk, hogy csak a felszíni izmok észlelhetőek. A mélyebb izmok esetében (amelyeket felszíni izmok vagy csontok borítanak) elengedhetetlen a finom huzal vagy tűelektróda. A legjobb esetben bármilyen elektródatípus szabad kiválasztását egy EMG- (elő)erősítő egészíti ki. Az elektróda típusának kiválasztása erősen függ az adott vizsgálattól és állapottól, egy elektródatípus nem képes minden lehetséges követelményt lefedni.

A felületi elektródáknál az ezüst/ezüst-klorid előzselésített elektródák a leggyakrabban használt elektródák, és általános használatra ajánlottak (SENIAM – felületi elektromyográfia az izmok non-invazív értékeléséhez). Emellett könnyű és gyors kezelés, a higiéniai szempontok nem jelentenek problémát, ha ezt az eldobható elektródatípust használják. Az elektródák általános átmérője (vezetőképessé metszetének területe) 1 cm vagy kisebb.

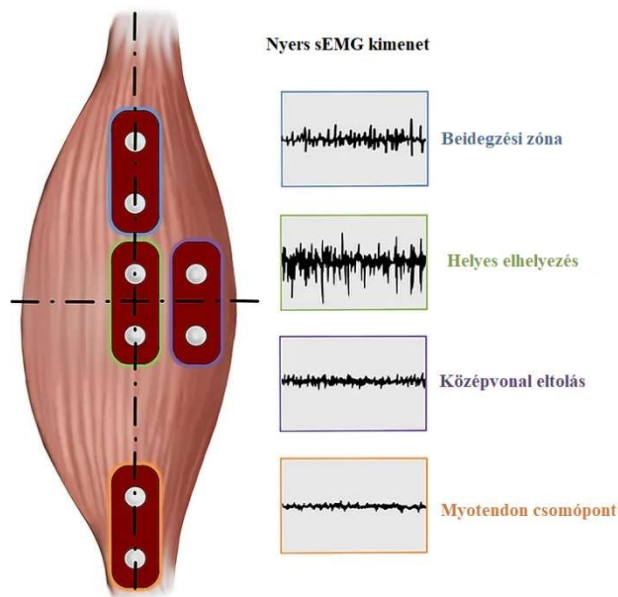
A kereskedelmi forgalomban kapható eldobható elektródákat nedves gélelektrodaként vagy ragasztógél elektródaként gyártják (11. ábra). Általában a nedves gél elektródák jobb vezetési és impedancia feltételekkel rendelkeznek, mint a ragasztógél elektródák. Ez utóbbinak az előnye, hogy hiba esetén áthelyezhető.



11. ábra: Speciális EMG elektródák (1,2: NORAXON INC. USA) és normál EKG elektródák (3,4: AMBU-Blue) (Peter, 2005)

A felületi elektromyográfiás (sEMG) jel az elektromos komplex interferencia mintázata az izomösszehúzódás során. Az amplitúdója általában 0,01 mV és 10 mV között van fő, energiájának frekvenciasávja 0 Hz és 500 Hz között koncentrálódik. Az sEMG jelek detektálása non-invazív módszer, ami nagyon fontos a klinikai diagnosztikában, a rehabilitációs gyógyászatban és az intelligens protézisvezérlésben. Az elmúlt években elterjedt még a használata játékvezérléshez is. A jelek megszerzése után nagyon fontos lépés a jelfeldolgozás. (Jiajia, Xiaoou, & Z. Jane, 2019)

Az elektródákat az izomrost központi régiójára kell felhelyezni (izomhas), hogy rögzíthető legyen annak teljes feszültsége. A teljes feszültség és az erősebb jel igen szükséges egy felületi elektromyográfiai vizsgálat során, hiszen így pontosabban rögzíthető az izom aktivitása. Az izomhasat, azaz az izom kiszélesedő részét úgy a legegyszerűbb megtalálni, hogy összehúzódás és ellazulás közben tapintjuk ki. A megfelelő felhelyezést a következő ábra mutatja (12. ábra).



12. ábra: Megfelelő szenzor felhelyezés az izomra (www.biosourcesoftware.com, 2022)

1.3. Az izomtevékenység felismerésére alkalmazható módszerek

Az izomtevékenység mérése és felismerése támaszkodik a matematika, a műszaki informatika, az elektronika és a mérés technika által már elért eredményekre. Másrészt olyan megoldási módszereket és eljárásokat kínál, amelyek más tudományokban, például az orvostudományban és a rehabilitációs technológiákban is alkalmazhatók. Az izomtevékenység felismerése jellegét és módszereit tekintve egyértelműen a mérnöki tudományok körébe tartozik. A digitális jelfeldolgozás szoros kapcsolatban áll a számítógépes mérés technikával, amelynek, célja, hogy – ebben az esetben az EMG adatokat – pontosan mérjük, tároljuk és feldolgozzuk. A mérés technikai rendszerek információkat gyűjtenek, megjelenítik és digitálisan feldolgozzák az adatokat, amelyeket ezután különböző mesterséges intelligencia algoritmusok segítségével lehet kategorizálni. (Kelemen, 2016)

Az EMG jel nem periodikus jel. Ezért nagyon fontos, hogy pontosan kinyerjük a jellegzetes információkat a felszíni EMG jelekből. Az utóbbi időben új jelfeldolgozási technikák és matematikai modellek teszik lehetővé, hogy fejlett EMG észlelési és elemzési módszereket fejlesszenek ki. Különböző matematikai és gépi tanulási technikákat, mint például mesterséges neurális hálózatokat (ANN-Artificial Neural Networks), fuzzy rendszereket, valószínűségi modellek algoritmusait, metaheurisztikus és rajintelligencia algoritmusokat, valamint néhány hibrid algoritmust alkalmaznak az EMG jelek jellemzésére. (Bekir, 2014)

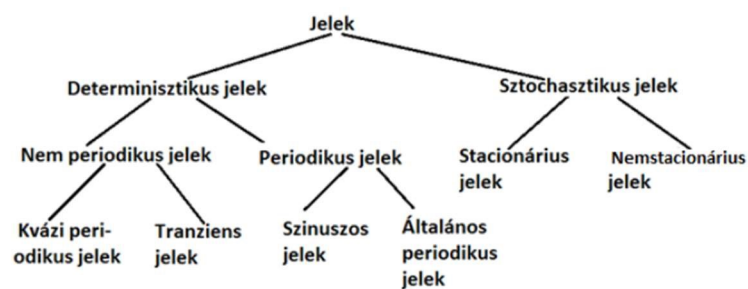
1.3.1. Jelkezelés és jelfeldolgozás

A jelek olyan fizikai mennyiségek, amelyek, egy adott fizikai rendszer állapotát írják le egy másik fizikai mennyiség, például az idő függvényében. A jelek vizsgálatakor általában egy független változót veszünk alapul, ez az értelmezési tartomány, amely meghatározza a jel viselkedését. A jel lehetséges értékeinek halmazát értékkészletnek nevezzük. A jelek osztályozásának két fő szempontja van az értelmezési tartomány és az értékkészlet.

Az analóg jelek, amelyek mind az értelmezési tartományukban, mind az értékükben folytonosak. Ezek a jelek a valós világ fizikai mennyiségeit folyamatosan változó formában reprezentálják.

Egy jelet diszkrét amplitúdójúnak nevezünk, ha az értelmezési tartománya folytonos, de az amplitúdó értékei csak meghatározott, különálló szinteken vehetnek fel értéket. Ebben az esetben a jel ugrásszerűen változik az amplitúdók között, míg a független változó továbbra is folytonos marad.

A következő ábrán (13. ábra) a jelek különböző típusainak osztályozása látható, amely alapján két fő kategóriát különböztethetünk meg: determinisztikus és sztochasztikus jeleket. A determinisztikus jelek további alcsoportjai közé tartoznak a **nem periodikus** jelek (amelyek lehetnek kvázi periodikusak vagy transziens jelek), valamint a periodikus jelek (ezek közül kiemelhetők a szinuszos és az általános periodikus jelek). A sztochasztikus jeleket pedig aszerint osztályozzuk, hogy sztochasztikus stacionárius vagy sztochasztikus nemstacionárius jelek. (Kelemen, 2016)

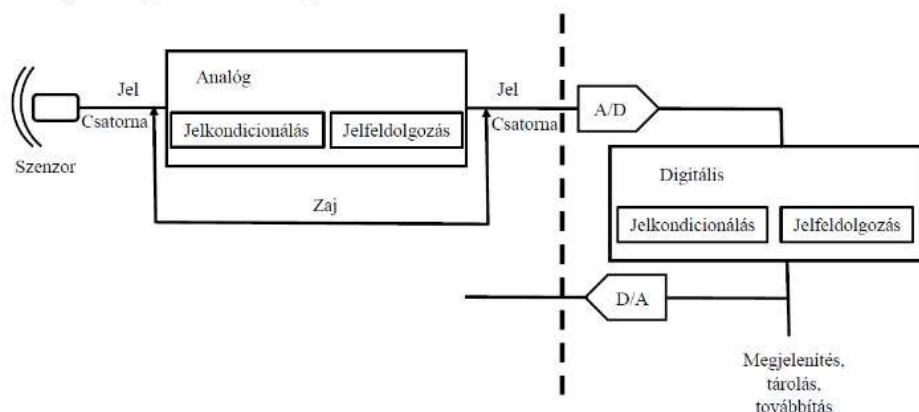


13. ábra: Jelek osztályozása (Kelemen, 2016)

Ha ezt az osztályozási rendszert alkalmazzuk az EMG (elektromyográfiás) jelre, azt sztochasztikus jelnek tekinthetjük, mivel az izomtevékenység során keletkező elektromos aktivitás változó, és nem határozható meg előre egzakt módon. Ezen belül az EMG jel nemstacionárius sztochasztikus jel, mivel az izomaktivitás során a jel statisztikai tulajdonságai

időben változnak. Az izomösszehúzódások és ellazulások miatt a jel amplitúdója és frekvenciataralma nem állandó, így nem tekinthető stacionáriusnak.

A jelfeldolgozás a mérési folyamatok alapvető és kulcsfontosságú része, amely számos területen elengedhetetlen, például folyamatirányítási rendszerekben, orvosi eszközökben, robotikában és mérőműszerekben. A jelek feldolgozása történhet analóg vagy digitális rendszerek segítségével. Korábban az analóg jelfeldolgozás dominált, de a digitális technológia fejlődésével a valós idejű feldolgozás egyre inkább digitális rendszerekben zajlik. Amikor analóg jeleket analóg eszközökkel dolgozunk fel, analóg jelfeldolgozásról beszélünk, míg a digitális jelek esetén digitális jelfeldolgozásról (DSP) van szó. Ha analóg jeleket szeretnénk feldolgozni, először azokat egy jelátalakítóval elektromos jellé kell alakítani. Ezután a jelet erősíteni kell, hogy továbbítható legyen, és zajszűrésre is szükség van, mert a jelek gyakran zajjal terheltek.



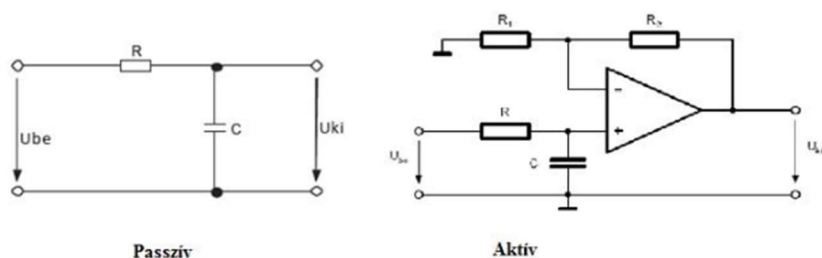
14. ábra: A jelfeldolgozás bloksémája (Sarcevic, 2019)

Amennyiben egy nem elektromos természetű jelet akarunk digitálisan feldolgozni, akkor a jelfeldolgozás legelső lépése ezt a jelet analóg elektromos jellé (feszültséggé) alakítani, amelyre különböző jelátalakítókat használnak. Az én esetemben erre nem volt szükség, így ez a lépés kimaradt a jelfeldolgozási folyamatból.

A jelkondicionálás folyamata során, még mielőtt a jel digitalizálásra kerülne, az elektromos jelet megfelelően erősíteni és szűrni kell. Az erősítésre azért van szükség, mert a jelátalakítók által kibocsátott jel gyakran túl gyenge ahhoz, hogy közvetlenül továbbítható legyen, különösen hosszabb kábelon keresztül. A szűrés két fő célt szolgál: egyrészt minimalizálja a zajt, amely zavaró frekvenciákat és jeleket tartalmaz, amelyek nem részei a feldolgozni kívánt hasznos információnak. A zaj csökkentése azért fontos, mert ezáltal növelhető a jel és a zaj aránya, amely meghatározza a jelfeldolgozás hatékonyságát. Másrészt

a szűrés egyik alapvető célja a jel spektrális átfedéseinek elkerülése, amelyek aliasing effektushoz vezethetnek. Az aliasing akkor alakul ki, ha a mintavételezési frekvencia túl alacsony, és emiatt spektrális átfedések jelennek meg, amelyek torzítják a jelet. Ennek elkerülése érdekében a Shannon–Nyquist mintavételezési tétel kimondja, hogy a mintavételezési frekvenciának legalább kétszer akkorának kell lennie, mint az analóg jel legmagasabb frekvenciájának. Ha ez a feltétel nem teljesül, a jel nem rekonstruálható teljes mértékben az összegyűjtött minták alapján, ami a mérési pontosság jelentős csökkenését eredményezheti.

Ezeknek a céloknak az elérésére gyakran alkalmaznak passzív vagy aktív aluláteresztő szűrőket (15. ábra), amelyek a kis frekvenciájú komponenseket változatlanul átengedik, míg a magasabb frekvenciákat jelentősen csökkentik. Ezzel biztosítják, hogy a feldolgozott jel csak a kívánt információt tartalmazza, csökkentve az esetleges torzításokat és zajokat.



15. ábra: Passzív és aktív aluláteresztő szűrők (Kelemen, 2016)

Az analóg-digitális átalakítók (A/D konverterek) a bemeneti feszültséget annak nagyságával arányos egész számmá alakítják. Az átalakítás pontosságát azonban több paraméter is befolyásolja:

- Referencia feszültség (U_{ref}): Az az érték, amely a maximálisan elfogadott analóg feszültséget jelöli.
- Teljes tartomány (Full Scale): A bemeneti jel azon intervalluma, amelyen belül az átalakító telítetlenül működik. Ezt néha vezérlési tartománynak is nevezik.
- Felbontás (Resolution): Megmutatja, hogy hány különböző binárisan kódolt érték jeleníthető meg a digitális oldalon. A felbontás mértékegysége a bit. Például egy 4 bites A/D átalakító 16 különböző értéket képes megjeleníteni.

A bemeneti feszültséghez rendelt digitális értéket a következő képlet adja meg:

$$Z = \left[\frac{U_{2^b}}{U_{ref}} + 0.5 \right] \quad (1)$$

- A legkisebb szignifikáns bit (LSB) azt jelzi, hogy mekkora bemeneti jelváltozás szükséges egy bitnyi változáshoz a digitális kimeneten. Ha az A/D átalakító felbontása N bit, akkor az LSB kiszámítása a teljes tartomány és felbontás alapján történik.
- Sáv szélesség: Az A/D konverter által elfogadott bemeneti jel frekvenciatartománya. Ez a tartomány határozza meg, hogy a bemeneti jelet milyen frekvenciákon tudja helyesen kezelni. A sáv szélességen kívül eső frekvenciákat az átalakító figyelmen kívül hagyja.
- Mintavételezési frekvencia: Meghatározza, hogy az A/D átalakító másodpercenként hányszor képes a bemeneti jelet digitalizálni. Mértékegysége a minták száma másodpercenként (SPS – Sample per Second).
- Konverziós idő: Az az időintervallum, amely szükséges egy minta átalakításához analógról digitálisra.

A mintavételezés során a folyamatosan érkező analóg jelből meghatározott időközönként (diszkrét időpontokban) veszünk mintákat, és ekkor rögzítjük az analóg jel pillanatnyi értékét. A digitális jelfeldolgozásban általában egyenletes mintavételezést alkalmazunk, ami azt jelenti, hogy az időtengelyt egyenlő távolságokra osztjuk, és ezekben a pontokban történik a mintavétel. Az $f(t)$ diszkrét mintaértékei egy f_n sorozattal adhatók meg, ahol T a mintavétel periódusideje.

$$f_n = f(t_0 + kT) \quad (2)$$

Miután megtörtént a jelek mintavételezése, a digitalizált jeleinket dolgozzuk fel valamilyen jelfeldolgozó eszköz segítségével. A mikrovezérlők (mikrokontrollerek) egyszerű valós idejű vezérlési feladatokra lettek kifejlesztve, és egyetlen integrált áramkörben tartalmazzák a mikroprocesszort, memóriát és perifériákat. Kis méretű memóriával rendelkeznek, és alacsony frekvencián működnek. Programozásuk C/C++ vagy Assembly (esetleg funkcionálisan korlátozott Python) nyelven történik, és a szekvenciális futás megbízható működést biztosít. Belső időzítők segítik a valós idejű működést, bár valódi operációs rendszer nem telepíthető rájuk, egyszerű valós idejű rendszerek, mint a FreeRTOS, használhatók. Támogatnak A/D és D/A átalakítást, valamint különböző kommunikációs interfészeket. Többféle mikrovezérlő típus létezik, például PIC, AVR, ARM Cortex-M, és 8,

16 vagy 32 bites változatok. Modern mikrovezérlők tartalmaznak lebegőpontos egységeket (FPU) is a gyorsabb jelfeldolgozáshoz.

A jelfeldolgozás során a cél a jelek szűrése, transzformálása, tömörítése vagy egyéb algoritmusok megvalósítása, amivel például kategorizálhatóak vagy felismerhetőek az általunk mért izommozgások. (Sarcevic, 2019)

1.3.2. Gépi tanulási technikák

A tanuló algoritmusok adaptív módszerek, amelyekben a hálózati számítási egységek önszerveződnek, hogy megvalósítsák a célt (vagy kívánt viselkedést). A gépi tanulás arról szól, hogy minták alapján megtanulunk előre jelezni célzott viselkedést vagy adatokat múltbeli megfigyelések alapján. A gépi tanulási algoritmusok a következőképpen osztályozhatók:

1. Felügyelt tanulás, ahol az algoritmus létrehoz egy olyan függvényt, amely az inputokat a célkimenetekhez rendeli. A tanuló ezután összehasonlítja a tényleges választ a célkimenettel, és úgy módosítja belső memóriáját, hogy legközelebb nagyobb valószínűséggel adjon megfelelő választ ugyanazon bemenetre.
2. Nem felügyelt tanulás (klaszterezés, dimenziócsökkentés, ajánlórendszerek, önszerveződő tanulás), amely az inputokat modellezi. Nincsenek célkimenetek (nincs címkézett példa). A tanuló nem kap visszajelzést a környezettől.
3. Félig felügyelt tanulás, ahol az algoritmus egy speciális függvényt hoz létre címkézett és nem címkézett példákból egyaránt.
4. Megerősítéses tanulás, amely során a tanulás a környezettel való interakció révén történik. A tanuló visszajelzést kap a válasza helyességéről.
5. Tanulás tanulása, ahol az algoritmus saját induktív előfeltételét (inductive bias) korábbi tapasztalatok alapján tanulja meg. Ezt induktív tanulásnak nevezzük.

(Bekir, 2014)

A **mesterséges neurális hálózatok** (Artificial Neural Network, ANN) ahogy a nevük is jelzi olyan számítási hálózatok, amelyek szimulálni a biológiai (emberi vagy állati) központi idegrendszer idegsejtjeinek (neuronjainak) hálózataiban zajló döntéshozatali folyamatokat. Ez a szimuláció egy bruttó sejtenkénti (neuron-neuron) szimuláció. A biológiai neuronok és azok neurofiziológiai hálózataiból szerzett tudás alapján működik. Így különbözik a hagyományos (digitális vagy analóg) számítástechnikai gépektől, amelyek az emberi agy számításainak helyettesítésére, javítására vagy felgyorsítására szolgálnak úgy, mint például egy egyszerű kézi számológép.

A mesterséges neurális hálózat, mivel egy biológiai neurális hálózatot szimulál, valójában egy új számítógép-architektúrát és algoritmizálási architektúrát jelent a hagyományos számítógépekkel szemben. Lehetővé teszi nagyon egyszerű számítási műveletek (összeadások, szorzások és alapvető logikai elemek) használatát olyan összetett, matematikailag túl komplex módon definiált, nemlineáris vagy sztochasztikus problémák megoldására, amelyekhez egy hagyományos algoritmus bonyolult egyenletrendszereket alkalmaz, és csak egy adott problémára lenne használható. A mesterséges neurális háló számítási és algoritmikai szempontból nagyon egyszerű, és önszerveződési tulajdonsággal rendelkezik, ami lehetővé teszi, hogy széles körű problémákra alkalmazható legyen.

Amelyben még a mesterséges neurális hálók eltérnek és potenciálisan előnyösek a hagyományos számítógépekhez képest, az a nagyfokú párhuzamosságuk (elemi szintű párhuzamosság). Nyilvánvalóan nem várható el, hogy az ANN mindent megtegyen a mostani számítógépek helyett, mivel kétségkívül még van hova fejlődniük. Mégis elmondhatjuk, hogy mára az ANN-k fontos szerepet játszanak a döntéshozatal, az információkeresés, az előrejelzés, a felismerés, a gépdiaosztika, az irányítás, az adatbányászat és kapcsolódó területek számos aspektusában, valamint ezek alkalmazásaiban az emberi tevékenységek számos területén. (Graupe, 2013)

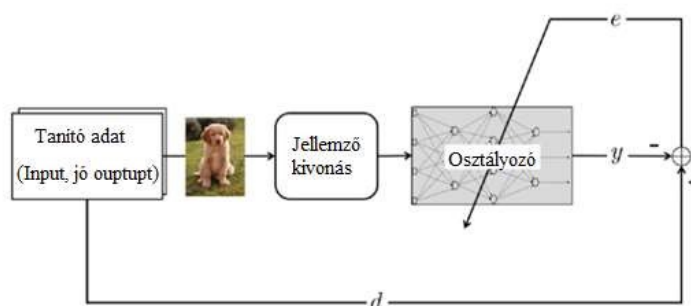
Konvolúciós neurális hálók (CNN):

A mély (több rétegű) neurális hálózat fontossága abban rejlik, hogy megnyitotta az ajtót a bonyolult nemlineáris modell és a szisztematikus megközelítés előtt a tudás hierarchikus feldolgozásához. A konvolúciós neurális hálózat (ConvNet) egy mély neurális hálózat, amely képfelismerésre specializálódott, viszont kiterjedt adatbázisok és idősorok mintázatfelismeréséhez is használható, ahogy azt én is használtam a tanulmányomban. Valójában a ConvNet egy régi technika, amelyet az 1980-as és 1990-es években fejlesztettek ki, egy időre azonban feledésbe merült, mivel nem volt praktikus valós idejű alkalmazásokhoz az akkori technikai határok (nem elégedő számítási kapacitás) miatt. 2012-ben azonban újra népszerű lett, és gyors ütemben hódított teret a neurális hálózatok módszertanában.

A ConvNet nem csupán egy mély neurális hálózat, amelynek sok rejtett rétege van. Ez egy olyan neurális háló, amely utánozza az agyunk látókéregben történő képfeldolgozását. Konceptiójában és működésében különbözik a korábbi neurális hálózatoktól, ezért fontos bemutatni az alapvető architektúráját.

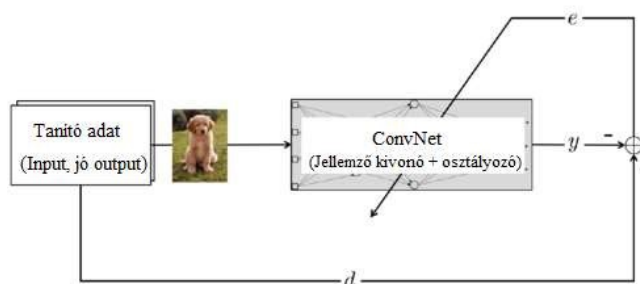
Alapvetően a képfelismerés az egy osztályozás. Például felismerni azt, hogy macskát vagy kutyát látunk, az agyunk segítségével osztályokba soroljuk őket, és a kutya megy a kutya

osztályba a macska pedig a macskák osztályába, mert az évek alatt megtanultuk ezeket elkülöníteni egymástól, holott sok fajtája van mindkettőnek. Ugyanez vonatkozik a betűfelismerésre vagy a mi esetünkben az izomműködési adatok felismerésére. Ezért a konvolúciós hálózat kimeneti rétege általában többosztályú besorolást alkalmazó neurális hálózat. A ConvNet előtt a jellemző kivonók (extractorok) szakemberek tervezték meghatározott területeken. Ez sajnos jelentős költséget és időt igényelt, és a teljesítménye is inkonzisztens volt. Ezek a jellemző kivonók (feature extractors), funkciókivonók függetlenek voltak a gépi tanulástól, ezt mutatja be a következő ábra (16. ábra).



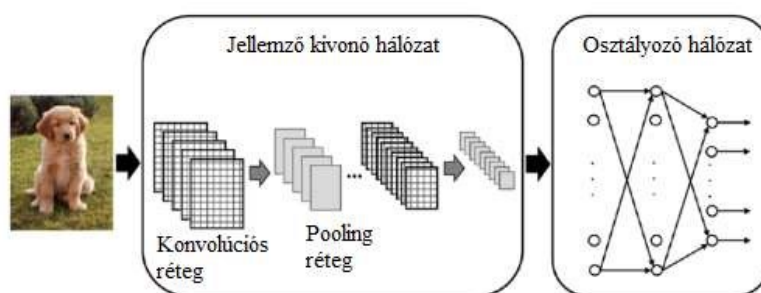
16. ábra: A jellemző kivonók függetlenek voltak a gépi tanító algoritmustól (Phil, 2017)

A ConvNet az adatokban lévő jellemzők kinyerését magába foglalja a tanítási folyamata során ahelyett, hogy azt manuálisan kellene megtervezni. A CNN jellemző kinyerője különleges típusú neurális hálóból áll, melyek súlyait a tanítási folyamat során határozzák meg. Tehát a konvolúciós hálózat fő jellemzője, hogy a kinyerésért felelős folyamatot automatizálja, ami a következő ábrán látható (17. ábra).



17. ábra: A konvolúciós hálóban az osztályozó és a jellemző kivonó összeolvad (Phil, 2017)

A konvolúciós hálózat jobb eredményt biztosít, ha a „feature extractor” neurális hálózat mélyebb (több réteget tartalmaz). Máshogy leírva a ConvNet tartalmaz egy neurális hálót, ami kivonja az adatok jellemzőit, és tartalmaz egy neurális hálót, amely osztályokat hoz létre ez alapján. Az általános architektúrája a ConvNetnek a következő képen látható (18. ábra).



18. ábra: Tipikus felépítése egy konvolúciós hálónak (Phil, 2017)

A bemeneti kép (vagy információ) belép a jellemző kivonási hálózatba. A kivont jellemzők továbbítódnak az osztályozási neurális hálózatba, a hálózat ezután a kép jellemzői alapján működik, és generálja a kimenetet.

A jellemzőkivonási neurális hálózat a konvolúciós halmaz rétegeiből és összevonó rétegpárokból áll. A konvolúciós réteg, ahogy a nevéből is adódik, az adatokat konvertálja, úgy is tekinthetünk rá, mint digitális filterek összesége. A pooling réteg a szomszédos pixeleket egyetlen pixellé alakítja (kép esetén), ezzel csökkentve az adatméretet. A konvolúció és a pooling rétegek egy kétdimenziós síkban helyezkednek el egymáshoz képest.

Összefoglalva, a ConvNet a jellemzők kinyerésére szolgáló hálózat és az osztályozó hálózat soros összekapcsolásából áll. A tréning folyamat során mindkét réteg súlyait meghatározzák. A jellemzők kinyerési rétege a konvolúciós és a pooling rétegek párba állításával működik. A konvolúciós réteg a konvolúciós művelet segítségével alakítja át a képeket, míg a pooling réteg csökkenti a kép dimenzióját. Az osztályozó hálózat általában hagyományos, többsztályos osztályozó neurális hálózatot alkalmaz. (Phil, 2017)

Support Vector Machine:

A Support Vector Machine (SVM) egy felügyelt tanulási algoritmus, amelyet osztályozásra és regresszióra is lehet alkalmazni. Az algoritmust statisztikai tanulási elmélet alapján Vapnik és Chervonenkis alkotta meg az 1960-as években. Az SVM egy nem-valószínűségi, bináris lineáris osztályozó, amely két kategória közötti különbséget határoz meg (de ki lehet terjeszteni több osztályra is). Az SVM tréning algoritmus egy modellt épít a tanító adatokból, amely az új példákat a két kategória egyikébe sorolja. Az adatok pontokként jelennek meg a térben, úgy, hogy az eltérő kategóriák példái között a lehető legszélesebb rés legyen. Az SVM pontjai a döntési felülethez legközelebb eső adatok. Az új példákat ezen térbe vetítik, és azok elhelyezkedése alapján sorolják be őket. Az SVM számos területen sikeres

alkalmazásokkal és jó teljesítménnyel rendelkeznek, például a bioinformatikában, szövegfeldolgozásban és mintafelismerésben. (Bekir, 2014)

Az SVM a strukturális kockázat minimalizálásán (SRM) alapul. Az SVM az input vektort egy magasabb dimenziós térbe képezi le, ahol egy maximális elválasztó hipersíkot hoz létre. Két párhuzamos hipersík kerül felépítésre a fő hipersík mindkét oldalán, amely elválasztja az adatokat. Az elválasztó hipersík az, amely maximalizálja a távolságot a két párhuzamos hipersík között. Feltételezik, hogy minél nagyobb a margó vagy a távolság ezek között a párhuzamos hipersíkok között, annál jobb lesz az osztályozó általánosítási hibája.

Vegyük figyelembe az alábbi adatpontokat a következő formában: $\{(x_1, y_1), (x_2, y_2), (x_3, y_3), \dots, (x_n, y_n)\}$, ahol $y_n = 1$ vagy -1 , egy konstans, amely azt jelzi, hogy az adott x_n pont melyik osztályba tartozik. n az adatminta száma, és minden x_n egy 3-dimenziós valós vektor. A skálázás fontos, hogy megakadályozzuk a nagyobb szórású változók (attribútumok) dominanciáját. A betanítási adatokat a következő módon szemléltethetjük az elválasztó (vagy osztó) hipersíkkal, amelyet az alábbi egyenlet ír le:

$$w \cdot x + b = 0 \quad (3)$$

Ahol b egy skalár és w egy p -dimenziós vektor. A w vektor merőlegesen mutat az elválasztó hipersíkra. A b eltolt paraméter hozzáadása lehetővé teszi a margin növelését. b hiányában a hiper-síknak át kellene haladnia az origón, ami korlátozná a megoldást. Mivel a maximális marginra törekszünk, az SVM és a párhuzamos hipersíkok érdekesek számunkra. A párhuzamos hipersíkok az alábbi egyenletekkel írhatók le:

$$w \cdot x + b = 1 \quad (4)$$

$$w \cdot x + b = -1 \quad (5)$$

Ha a tanító adatok lineárisan szeparálhatók, akkor úgy választhatjuk meg ezeket a hipersíkokat, hogy közöttük ne legyenek pontok, és megpróbáljuk maximalizálni a távolságukat. A geometria alapján a hipersíkok közötti távolság $2/|w|$ (19. ábra). Ezért azt szeretnénk, hogy $|w|$ minimalizálva legyen. Az adatpontok kiválasztásához biztosítanunk kell minden egyes i -re hogy:

$$w \cdot x_i + b \geq 1 \quad (6)$$

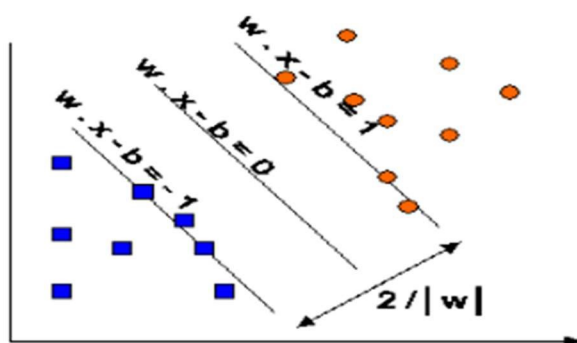
vagy

$$w \cdot x_i - b \leq -1 \quad (7)$$

A fenti két egyenlet átírható a következő formába:

$$y_i(w \cdot x_i - b) \geq 1 \text{ ahol } 1 \leq i \leq n \quad (8)$$

Itt y_i a címke, amely értéke lehet 1 vagy -1, attól függően, hogy az adott adatpont melyik osztályba tartozik. (Himani & Mahesh H., 2012)



19. ábra: Maximális marginális hipersíkok egy két osztályból származó mintákkal betanított SVM-hez (Himani & Mahesh H., 2012)

K-Leközelebbi Szomszéd (KNN):

A K-nearest neighbour (KNN) osztályozási szabály az egyik legismertebb és legszélesebb körben használt nemparaméteres mintafelismerési módszer. Ezt eredetileg Cover és Hart javasolta, és egyszerűsége, valamint hatékonysága miatt számos osztályozási problémában használják. Amikor csekély vagy semmilyen előzetes információ nincs az adatok eloszlásáról, a KNN módszer a kedvelt választások közé tartozik osztályozásnál. Az KNN egy „lustán” tanuló módszer, mert a tanító adatokat nem vetik alá előfeldolgozásnak. Az adott minta osztálya a rendszer által ismert legközelebbi mintához tartozó osztály lesz, amit egy távolságmérték határoz meg a jellemzőtérben (attribútumtérben). Ebben a térben minden minta meghatároz egy régiót, amit Voronoi-régióknak nevezünk. Ha a távolság a klasszikus euklideszi távolság, akkor a Voronoi-régiókat lineáris határok választják el. Az 1-NN osztályozásnál jobbat elérhetünk azzal, hogy több szomszédot veszünk figyelembe egy minta osztályozásának meghatározásához (KNN), vagy az euklideszi távolság helyett más (Hamming, Koszinusz) távolságokat használunk. A KNN szabály úgy osztályozza x -et, hogy a K legközelebbi minták közül a leggyakrabban előforduló címkét rendeli hozzá; ez azt jelenti, hogy a döntést a K legközelebbi szomszéd címkéinek vizsgálata és azok többségi szavazata alapján hozzák. A KNN kifejlesztése abból a szükségletből fakadt, hogy diszkriminanciaelemzést lehessen végezni, amikor a valószínűségi sűrűségek megbízható paraméteres becslései ismeretlenek, vagy nehezen meghatározhatók. Manapság ez a leggyakrabban használt osztályozási algoritmus. Hátránya az, hogy kevésbé hatékony és munkaigényes, ha a tanító adathalmaz nagy.

2. ANYAGOK ÉS MÓDSZEREK

A célkitűzésem egy olyan adatgyűjtő rendszer megvalósítása volt, amely az EMG szenzorok alkalmazásával képes különböző izomcsoportok elektromos aktivitásának mérésére. További célom volt, hogy az így nyert mérési adatok elemzéséhez különböző gépi tanulási algoritmusokat próbáljak ki, amelyek segítségével az izomtevékenységeket kategorizálhatom és értékelhetem, ezzel olyan módszertanok kipróbálása, amely hozzájárulhat az izomcsoportok aktivitásának pontosabb megértéséhez és a kapcsolódó alkalmazások fejlesztéséhez.

2.1. Adatgyűjtő rendszer megvalósítása

A munkám során sEMG jeleket mértem és dolgoztam fel, melyek bioelektromos jelek, továbbá folyamatos analóg jelek, amik a bőr felszínén rögzített mikrovolt nagyságrendű elektromos potenciálváltozások formájában jelentkeznek. Ezen jelek alacsony frekvenciájúak és alacsony amplitúdójúak. Mivel az izomaktivitás intenzitás és mintázata az idő függvényében változik, ezért tipikus időbeli jel. Gyakran zajosak ezek a jelek, mivel tartalmazhatnak interferenciát más elektromos forrásból (számítógép, mikrovezérlő, hálózati zaj), ezért szűrni is kell őket a hasznos jelek kinyeréséhez. Jellemzően nem determinisztikusak, azaz nem lehet pontosan megjósolni az időbeli mintázatukat, mivel a jel erősen függ a mért izom állapotától és a mozgás típusától. Bár a méréseket digitálisan végeztem, maga a jel természetes formájában folytonos.

A fenti tényezőkhez alkalmazkodva kellett megvalósítanom egy olyan adatgyűjtő, feldolgozó rendszert, amely ezen jeleket megfelelően lekezeli, és a tanulóalgoritmusok számára már optimális formában adja tovább.

2.1.1. EMG szenzorok

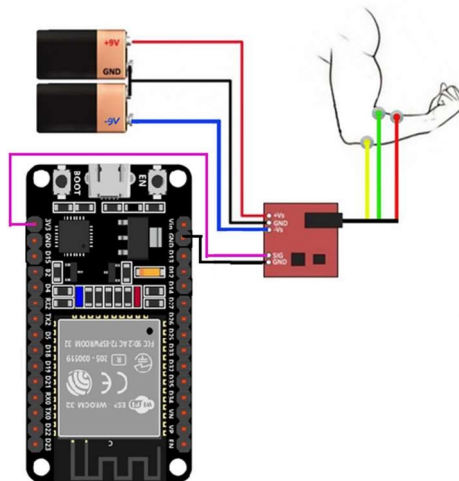
Munkám során kettő különböző, eltérő kategóriájú szenzort is kipróbáltam több kevesebb sikerrel.

Az első az egy kínai általános, hobbi projektekhez való szenzor volt 5 csatlakozási ponttal a mikrovezérlőhöz és egy 3.5mm-es jack csatlakozóval az izmokhoz (20. ábra). A bekötéséhez egy pozitív és negatív tápegység-, egy föld az elemhez-, egy analóg- és egy föld a mikrovezérlőhöz- csatlakozást kellett használni. Az izmokhoz három tapaszos csatlakozóra volt szükség, az egyiket a mért izomcsoport eredéséhez, a másikat a mért izomcsoport közepre és az utolsót pedig a működtetett izom közelébe egy csontos helyre kellett felhelyezni. (Shojaei, dátum nélk.)



20. ábra: Felület EMG tapaszos elektródákkal - (Shojaei, dátum nélk.)

A szenzor áramellátását kettő elemmel kellett biztosítani, így tudtam előállítani a $+V_s$ -t és $-V_s$ -t, tehát azt, hogy a rendszerünk kétpólusú legyen. A bekötést a következő ábra (21. ábra) szemlélteti.



21. ábra: ESP és EMG szenzor bekötési rajza

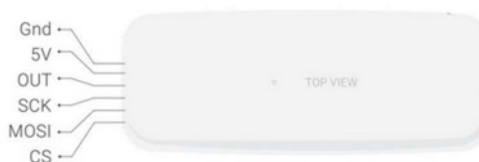
A másik EMG szenzor, amit használtam, az az Elemyo által gyártott MYO v1.4-es szenzor (22. ábra),

ami már egy magasabb kategóriás műszer fém érintkezési elektródákkal, viszont még mindig nem professzionális orvosi műszernek minősül.



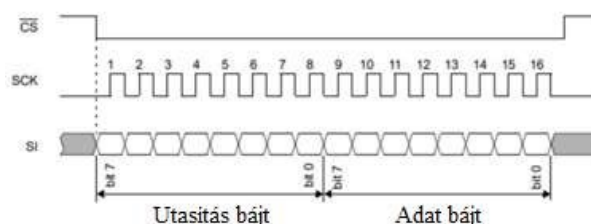
22. ábra: Elemyo v1.4 szenzor (www.elemyo.com, dátum nélk.)

A normál tápfeszültsége 5 V és alacsony áramfelvételű (3 mA). Az analóg kimeneti csatornáján az alapjelet (az alsó ábrán: OUT) tudja erősíteni x1-x32 szorzására, ami SPI (Serial Peripheral Interface) interfészen keresztül állítható. A csatlakozó kiosztása a következő (23. ábra):



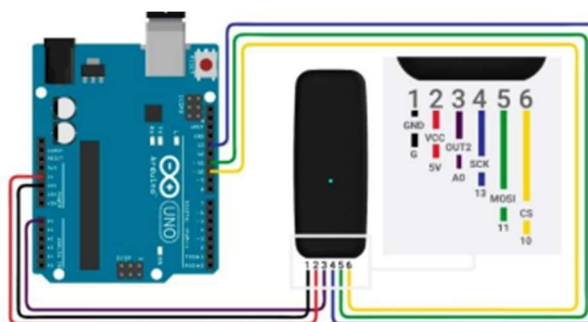
23. ábra: Elemyo v1.4 szenzor lábkievezetései (www.elemyo.com, dátum nélk.)

Az érzékelő erősítésének beállításához szabványos SPI-kompatibilis soros interfészt használ. Az MCU alacsonyra húzza a Chip Select (CS) lábat, hogy kommunikációt kezdeményezzen az érzékelővel. Minden SI utasítás első byte (két byte hosszú) az utasító byte, amely az ehhez tartozó regiszterben van elhelyezve. A parancsregiszter jelzi a második byte (Data byte) célállomását. Egy utasítás után, ami két byte hosszú a CS magasra vált. Az interfész működési elvét a következő ábrán láthatjuk (24. ábra):



24. ábra: Az interfész működése (www.elemyo.com, dátum nélk.)

A mikrovezérlőhöz, (ami ebben az esetben egy Arduino Uno volt) a következőképpen kellett bekötni (25. ábra):



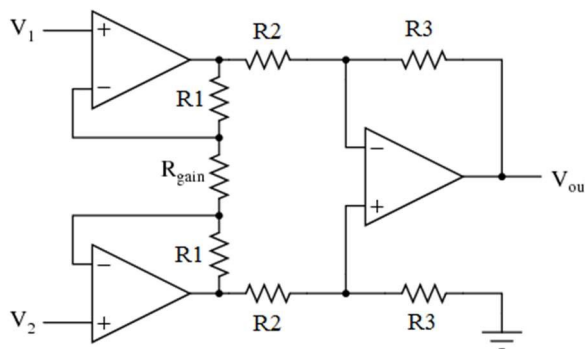
25. ábra: Arduino és Elemyo szenzor bekötése (www.elemyo.com, dátum nélk.)

Leszámítva, hogy milyen szenzort használunk mindegyik hasonló alapon működik, differenciális erősítési technikával. A differenciál erősítőknek magas bemeneti impedanciával és nagyon alacsony kimeneti impedanciával kell rendelkeznie, minden mérésnél erre szükséges törekedni. Ideálisan egy differenciál erősítőnek végtelen bemeneti és nulla kimeneti impedanciája van.

A differenciális erősítést egy műszererősítő segítségével érik el, amely magas bemeneti impedanciát biztosít. Egy általános három erősítős műszererősítő látható a következő ábrán (26. ábra), ehhez hasonlóan épülnek fel az általam használt szenzorok is. A műszerben lévő erősítő differenciális erősítést végez úgy, hogy kivonja a V_1 és V_2 feszültségeket. Így a V_1 és V_2 -n (elektróda bemenetei) közös zajjelek eltávolításra kerülnek. A differenciális erősítés mindkét bemeneten közös jelek elutasítására való hajlamát a közös módú elnyomási arány (CMRR) határozza meg. A 90 dB-es CMRR elegendő a közös jelek eltávolításához a műszererősítők esetében, de a legújabb technológia, bár drágább, de már 120 dB-es CMRR-t is biztosít. A műszererősítő erősítését egyetlen ellenállás (R_{gain}) segítségével állíthatjuk be. Az erősítési és a kimeneti egyenlet a műszererősítőre a következő egyenletekben látható. (Jamal, 2012)

$$Erősítés = \left(1 + \frac{2R_1}{R_{gain}}\right) \frac{R_3}{R_2} \quad (9)$$

$$V_{out} = (V_2 - V_1) * Erősítés \quad (10)$$



26. ábra: Három erősítős differenciális erősítés (Jamal, 2012)

2.1.2. Mikrovezérlők

Két mikrovezérlőt is használtam, az egyik egy ESP32, a másik pedig egy Arduino Uno volt.

Az ESP32 egy sokoldalú mikrovezérlő 2 magos processzorral rendelkezik, ami gyors adatfeldolgozást biztosít, és különböző perifériákat támogat például az SPI, I2C és UART

interfészeket, így számos külső eszközzel integrálható esetben a kínai EMG szenzorral. Az ESP32 továbbá hasznos valós idejű adat megfigyelés során. 512 KB rammal rendelkezik, 30-38 tű kiosztással és az óra frekvenciája 240 MHz.

Az Arduino Uno pedig egy ATmega328P-n alapuló mikrovezérlő kártya. 14 digitális bementi/kimeneti érintkezővel rendelkezik (ebből 6 PWM kimenetként használható), 6 analóg bemenettel, 16 MHz-es kerámia rezonátorral, USB csatlakozással, tápcsatlakozóval, ICSP fejléccel és reset gombbal. Mindent tartalmaz, ami a mikrokontroller támogatásához szükséges. Tápellátását 7 és 20 V közötti feszültséget fogadó hordócsatlakozó szolgálja.

Összességében egy nagyon felhasználóbarát és egyszerű megoldást biztosít széleskörű projektek elkészítéséhez és azért esett erre a választásom mert az EMG szenzorok többségének kompatibilitási szempontból az Arduino a legmegfelelőbb választás. (www.docs.arduino.cc, dátum nélk.)

2.1.3. USB leválasztó

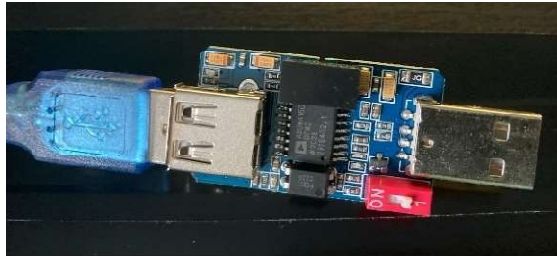
Amire még szükség volt az egy elszigetelő USB leválasztó. Ezt arra tervezték, hogy védelmet biztosítson az USP-perifériák túlfeszültség és az elektrosztatikus feltöltődéssel szemben az USB-porton. További szerepe a csökkentett hálózati interferencia az érzékelőrendszerek (EMG szenzor) csatlakoztatásakor olyan eszközökhöz, ami külső áramforráshoz van csatlakoztatva (szenzorunk csatlakoztatása a géphez), valamint kiegészítő áramütés elleni védelem.

Számos tényező vezethet veszélyes töltés vagy feszültség megjelenéséhez a tápegységhez csatlakozott eszközökön, amely az USB periféria meghibásodásához is vezethet, és sEMG szenzor esetén, ami érintkezik a testünkkel, áramütést eredményezhet.

Például veszélyt jelenthet, ha a számítógép tápellátási rendszere meghibásodik, vagy a földelés hiánya és a felhalmozódott statikus töltés a tokozáson is hibaforrás. Tehát az USB leválasztó kiküszöböli az USB elektromos meghibásodásának lehetőségét, és biztonságosabbá teszi a perifériák csatlakozását.

Az USB leválasztó (27. ábra) az ADUM3160-on alapul, ami egy szigetelt, digitális adatátviteli egység. Úgy mond izolált gate (kapu) meghajtó vagy izolált digitális izolátor, amely lehetővé teszi a digitális jelek szigetelt átvitelét két elkülönült rendszer között. Ezen kívül az általam használt izolátor magas és alacsony jelsebességet is tudott biztosítani (1.5 Mbps / 12 Mbps), ami különösen jó, mivel a gyors izomreakcióknál szükséges a nagyobb adatátviteli sebesség. Olyan területeken szükséges alkalmazni, ahol fontos az adatbiztonság és az elektromos zajoktól való védelem. Az izolált átvitel lehetővé teszi a digitális jelek átvitelét

anélkül, hogy galvanikus kapcsolatot eredményezne a két rendszer között, így minimalizálva a földhurok okozta zajt és a földhurok által továbbított potenciális zavarokat.



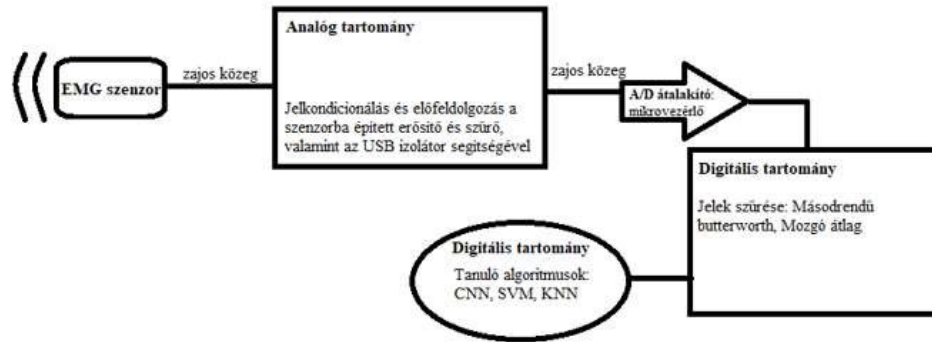
27. ábra: USB leválasztó

Ahogy az ábrán is látszik (34. ábra), sok felesleges zajtól lehet megszabadulni egy egyszerű leválasztó használatával biztonságosabbá téve a rendszerünket. Ez a későbbiekben is hasznos volt, mert a mért jelek kevesebb zajjal érkeznek az adatfeldolgozó rendszerbe, továbbá megvédi a mikrovezérlőket.

2.1.4. Adatgyűjtő és feldolgozó rendszer felépítése

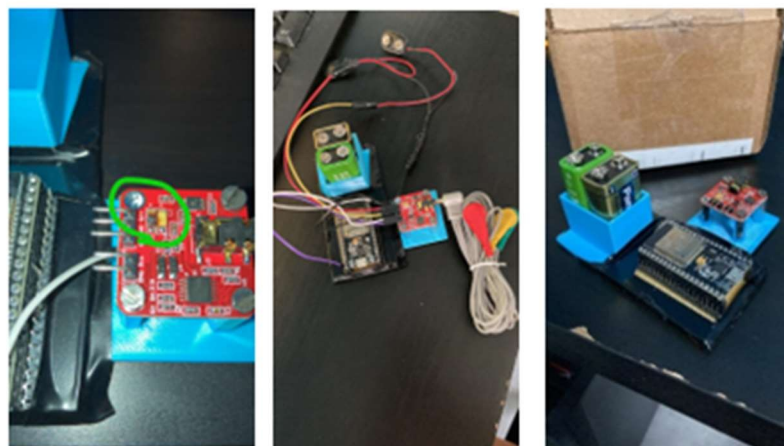
Miután kiválasztottam az adatgyűjtő rendszerem elemeit, meg kellett terveznem az adatgyűjtési folyamat blokkvázlatát, amely az egyes részegységek működését és kapcsolódását szemlélteti. Az elektromiográfiás jel mérése egy felületi EMG szenzorral kezdődött, amellyel az izmok elektromos aktivitását mértem. A jel elsőként áthalad egy analóg jel kondicionáláson és jelfeldolgozáson, amit ez esetben a szenzor és az USB izolátor biztosít. Ezt követően a jel átkerül a mikrovezérlőbe, ahol analóg-digitális átalakítás történik, digitalizálva a jelet további feldolgozáshoz. A digitalizált jel a digitális jelfeldolgozási folyamaton megy keresztül, amely magában foglalt egy másodrendű Butterworth szűrőt, amellyel a digitális jeleket szűrtem, ezen szűrő jelentőségét a későbbiekben részletezem. Végül a feldolgozott EMG jeleket különböző gépi tanuló algoritmusok (nevezetesen CNN, SVM, KNN) segítségével elemeztem, amelyek osztályozzák az izomcsoportok tevékenységeit.

A digitalizálás után már nem volt szükség a mikrovezérlőre, hanem számítógépes szoftverekkel oldottam meg a digitális szűrést és a tanító algoritmusok megvalósítását. Összegezve a fentieket a végső adatgyűjtő és feldolgozó rendszerem blokksémája a lenti ábrán (28. ábra) látható módon alakult.



28. ábra: Adatgyűjtő és feldolgozó rendszer bloksémája

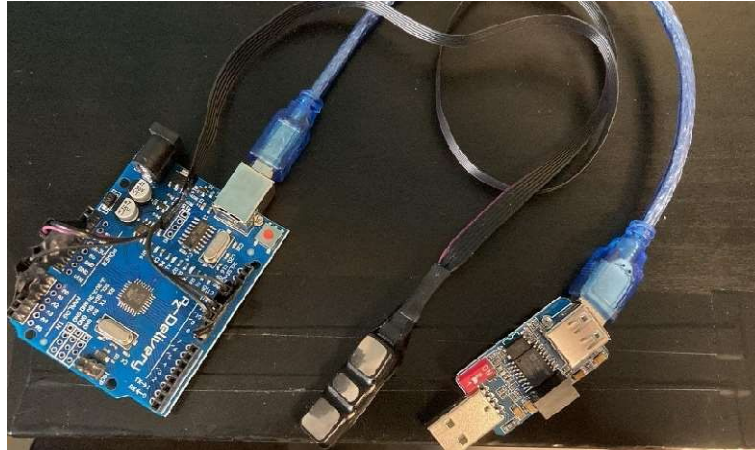
A valóságban pedig kettő adatgyűjtő rendszert valósítottam meg, ahogy a fentiekből is következik, az EMG szenzorokból és a mikrovezérlőkből is kettőt használtam fel. Első próbálkozásra az összerakott „adatgyűjtő állomás” egy ESP32 mikrovezérlőből és egy olcsó kínai EMG szenzorból állt (29. ábra). Ahhoz, hogy elkerüljük a földhurkot, külön tápra volt szükség a szenzor üzemeltetéséhez, amit két darab 9V-os elem segítségével oldottam meg. Ezeket úgy kellett bekötnöm, hogy legyen egy pozitív és egy negatív potenciál is, ami alapján majd feszített és a nem feszített izommozdulatot a szenzor el tudja különíteni. Sajnos egy hibás bekötést követően az EMG boardon kiégettem a kondenzátorokat, és azok kicserélése után sem tudtam már ezt a rendszert használni, mivel az ESP chipjén belül is valami összeégett (29. ábra -zölddel jelölt), így mindig brownoutra fut a működés a szenzor használata során, továbbá maximum potenciál jelenik meg a kimeneten.



29. ábra: ESP32 és kínai EMG szenzor „adatgyűjtő hardver”

A második „adatgyűjtő állomásom” szerencsére azóta is működik, és sikeresen le tudtam mérni vele a kívánt izomcsoportokat. Ezt már egy drágább szenzorból raktam össze,

amivel az Elemyo cég támogatott (www.elemyo.com, dátum nélk.). Ez az Elemyo v1.4-es felületi szenzor volt, és mellé egy Arduino Uno-t használtam, mint mikrovezérlő. Ahhoz, hogy elkerüljem a földhurkot és védjem az Arduino mikrovezérlőmet, ennél a rendszernél már használtam az USB-izolátort is. Ez lett a végső adatgyűjtő rendszerem, ami már megbízhatóbb volt és biztonságosabb (30. ábra), így ezt tudtam használni a hátralévő adatgyűjtéseim során.



30. ábra: Arduino Uno, Elemyo v1.4 és USB izolátor „adatgyűjtő rendszer”

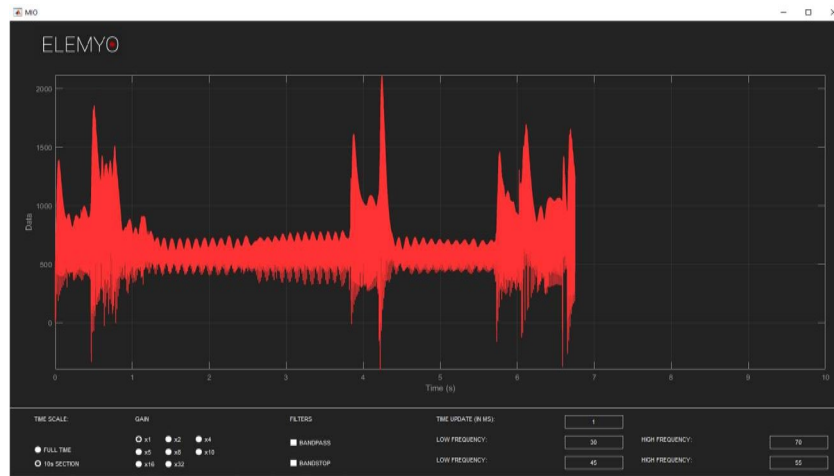
2.1.5. *Szoftverek*

A dolgozatom során több szoftvert is kipróbáltam, de végül két szoftver segítségével sikerült megvalósítanom a jel kezeléshez és -feldolgozáshoz, valamint mesterséges neurális hálók tanításához szükséges feladatokat. Az egyik a MATLAB volt, a másik pedig Visual Studio Code (VSC) Python.

A MATLAB egy interaktív környezet, amely alkalmas algoritmusok fejlesztésére, adatfeldolgozásra, szimulációk és vizualizációk elvégzésére is. Széles körben használják mérnöki alkalmazásokban, különösen a jel- és képfeldolgozásban, ezért is esett rá a választásom. Ezen program használatával mértem le a jeleimet, és alkalmaztam rajtuk utólagos szűrést.

Az Elemyo szenzorhoz a gyártó által készített és nyílt forráskódú scriptet használtam fel, amit kiegészítettem olyan funkcióval, hogy adatokat tudjak rögzíteni vele. A rögzíteni kívánt adatokat a program .csv formátumba mentette ki, amit utána Excel formátumba exportáltam, és a későbbiekben ezen adatokkal dolgoztam tovább. A gyártó által készített kód ugyan tartalmaz szűrőket, de ezeket nem használtam fel, szerencsére van opció ezen szűrők kikapcsolására. Azért volt előnyös, hogy ezt a scriptet használtam, mert ehhez már készült GUI (grafikus felhasználói felület) (31. ábra), így minden mérésem előtt le tudtam ellenőrizni, hogy

a szenzorjeleim megfelelőek, és hogy a szenzor megfelelően van-e beállítva és felhelyezve az adott bőrfelületre.



31. ábra: Nyers EMG jel, Elemyo GUI használatával

Ezen kívül a mért jeleimen a szűrők megfelelő beállítását is MATLAB segítségével végeztem. Egy egyszerű szűrő kódot készítettem, ami az általam kiválasztott szűrőket alkalmazza a kívánt EMG jel-en, egymás alá kirajzolva így tudtam megtalálni a megfelelő beállítást a szűrőimen (35. ábra). Ezeket az elkészített szűrőket (ld. később) pedig később a rögzített adatsoraimon alkalmaztam, de ezt már nem MATLAB segítségével végeztem el.

A tényleges szűrést és a tanuló algoritmusokat Python használatával valósítottam meg Visual Studio Code segítségével. Ezekről a későbbi fejezetekben kitérek.

2.2. Mérési adatgyűjtés és adatfeldolgozás

2.2.1. Adatgyűjtési módszerek

Annak érdekében, hogy az adataim a lehető legpontosabbak és legjobbak legyenek néhány általános megfontolást figyelembe kellett vennem, ami javítja a myoelektromos jelek mérését.

Mivel az sEMG mérés minősége nagymértékben függ a bőr megfelelő előkészítésétől és az elektródák pozicionálásától, ezért ügyeltem arra, hogy a bőrfelületeket igényesen előkészítsem. A bőr előkészítésének fő módja a stabil elektróda érintkezés és az alacsony bőr impedancia. Ugyan általános szabályok nincsenek rá, de számos praktika van, amivel javítani lehet az EMG-mérésekhez a szenzor terület előkészítését. Ha például kissé statikus vagy lassított mozgást tervezünk, akkor elegendő lehet egy egyszerű alkoholos tisztítás, de ha gyorsabb izommozgás közben szeretnénk mérni, akkor elengedhetetlen az alapos előkészítés. A következőket alkalmazva növeltem a bőrfelületem impedanciáját, a megfelelő mérésekhez.

1, A szőr eltávolítása: erre azért volt szükség, hogy javítsam az elektródák tapadását, nedves, izzadékonyabb bőrfelület esetén.

2, Bőrradírral, hogy eltávolítsam az elhalt hámsejteket. Tiszta alkoholt öntve durva törölközőre, és ezzel a bőrt radírozva le tudtam dörzsölni a felső elhalt réteget.

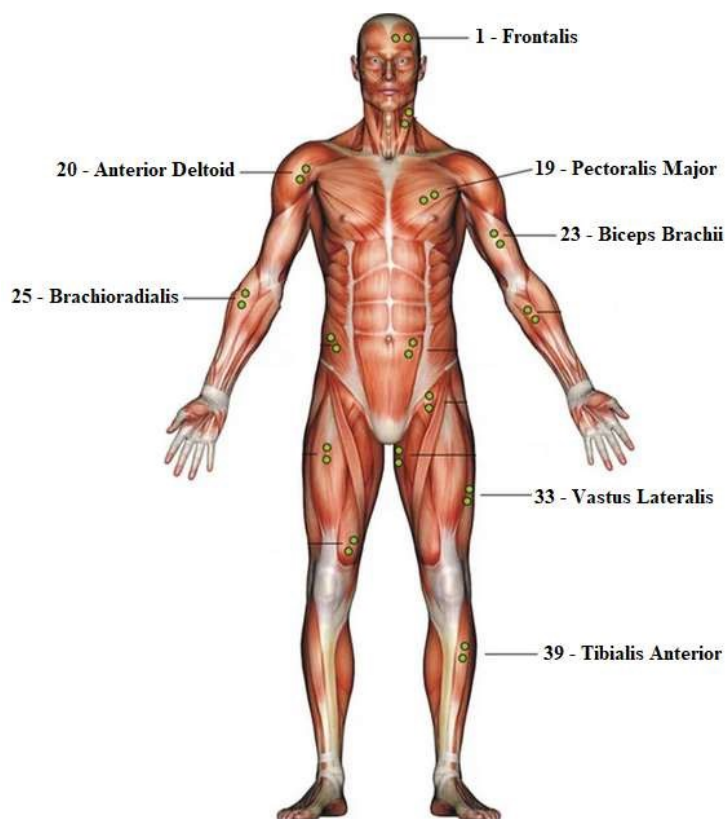
Annak érdekében, hogy még jobb eredményeket érjek el, a mérések közben a szenzor pontos felhelyezésére is nagyon oda kellett figyelni. A felületi EMG szenzor elektródákat úgy kell felhelyezni, hogy a motoros egység és az izom-ín kapcsolata között feküdjenek fel a bőrfelületre az izomhoz képest hosszában. Továbbá arra is figyelni kellett, hogy az adott izomhasra helyezzem fel a szenzort, mivel itt a legnagyobb az izomsűrűség, ez által javítva a jelerősséget.

A szenzor mivel nem öntapadó volt, hanem fémfelülettel van ellátva, ezért fel kellett ragasztani a bőrfelületre. Erre a legjobb megoldást egy kineziológiai szalag szolgáltatta, ami azért előnyös, mert mint egy „külső bőrként” követte az izomfeszülések által okozott mozgást, együtt nyúlva a tényleges bőrfelülettel (32. ábra). Ebből adódóan a szenzor megfelelően tudott felfeküdni a megjelölt területre.



32. ábra: Saját mérés Elemyo szenzor segítségével

A fenti lépéseket szükséges volt minden kijelölt mérési ponton elvégezni. Miután a megfelelő előkészületeket elvégeztem, kezdődhetett a mérés. Hét különböző izomcsoportot választottam ki, véletlenszerűen olyanokat, amiket az általam használt egyszerű szenzorral könnyű lemérni. Azt, hogy melyek voltak ezek az izomcsoportok, a következő ábra (33. ábra) szemlélteti. Minden izomcsoport mellé egy számot rendeltem, hogy a neurális hálók majd ez alapján tudják osztályozni az izomcsoportokat. Az izomcsoportok jelöléséhez az egyszerűség kedvéért számokat használtam.



33. ábra: Mért izomcsoportok (számozás és elnevezés) (www.biosourcesoftware.com, 2022)

Az izomcsoportok mérésénél nem határoztam meg konkrét mozgásokat, mivel nem az volt a cél, hogy mozgásformákat különítsek el, hanem az egyes izomcsoportokat, ezért minden izomnál csak egy egyszerű feszítés és ellazítás tett ki egy „adatcsomagot”. Természetesen ezt a lehető legtöbbször kellett megismételni, hogy a tanító algoritmusoknak minél több legyen az inputjuk. Izomcsoportonként 40-50 mintát rögzítettem.

A méréseket kizárólag saját magamon végeztem, részben a rendelkezésre álló korlátozott erőforrások miatt, részben pedig azért, mert így egy kezdeti kísérlet során jobban kontrollálható volt az adatgyűjtési folyamat. Egy nagyobb mintaszámú kutatás esetében, ahol több résztvevőn kellene mérni ugyanazon izomcsoportok aktivitását, számos változó jelenne meg, amelyek bonyolítanák az adatok elemzését, és az algoritmusok tanítását. Az emberi izomcsoportok aktivitása jelentős egyéni különbségeket mutat, például a testalkat, edzettségi szint és fáradtsági állapot függvényében.

Az ilyen típusú kutatásokhoz általában nagy számú résztvevő szükséges, hogy az adatok statisztikailag relevánsak és általánosíthatóak legyenek. Jelenlegi munkám során nem álltak rendelkezésre ilyen mértékű erőforrások, és az egyszerű otthoni szenzorokkal technikai és logisztikai szempontból is nehézkes lett volna több alany méréseinek begyűjtése. Mind emiatt

az adatok gyűjtésének optimalizálása érdekében a mérési folyamatot egyéni szinten tartottam, hogy az algoritmusok finomhangolására fókuszálhassak.

A mérések időtartamát próbáltam izomcsoportonként egyenletesen eloszlatni. Ez annyit jelent, hogy minden mérésnél egy feszítés és ellazítási ciklus volt egy „adatsomag”, ez nagyjából 1-3 másodperces ablakot jelent. Ennyi idő alatt már kirajzolódtak az adott izomcsoport karakterisztikái, amit a későbbiekben már a tanító algoritmusok fel tudtak ismerni. Természetesen a lehető legtöbbet kell mérni ilyen kis „adatsomagokból”, viszont az SVM-et és a KNN-t azért választottam, mert azok tanításához nem szükséges nagy mennyiségű adathalmaz.

Továbbá minden mérésemnél próbáltam rekonstruálni ugyanazt a környezetet. A szenzor felhelyezését a meghatározott pontokra, a testtartásomat úgy beállítani, hogy csak a mért izomra tudjak koncentrálni, és az feszüljön úgy, ahogy elvárt. Ezen kívül minden mérésem előtt megvizsgáltam az Elemyo GUI segítségével, hogy mindent jól állítottam-e be, és ha igen, akkor kezdtem csak el a méréseket.

A méréseim során a mintavételezési frekvenciát 1000 Hz-re állítottam, mivel ez elegendő volt a számomra releváns izomaktivitási jelek pontos rögzítéséhez, beleértve az emberi izomtevékenységhez kapcsolódó frekvenciatartományt. Ezzel a mintavételezési frekvenciával biztosítva volt, hogy a Shannon-Nyquist-tétel teljesüljön, és elkerüljem az aliasing jelenségét.

2.2.2. Adatfeldolgozás

Amikor EMG adatokkal próbálunk tanítani gépi tanítási modelleket, fontos alkalmazni különböző filtereket, és eldönthetjük, hogy ezt adatrögzítést közben vagy utána végezzük el, mindegyiknek megvan az előnye és a hátránya.

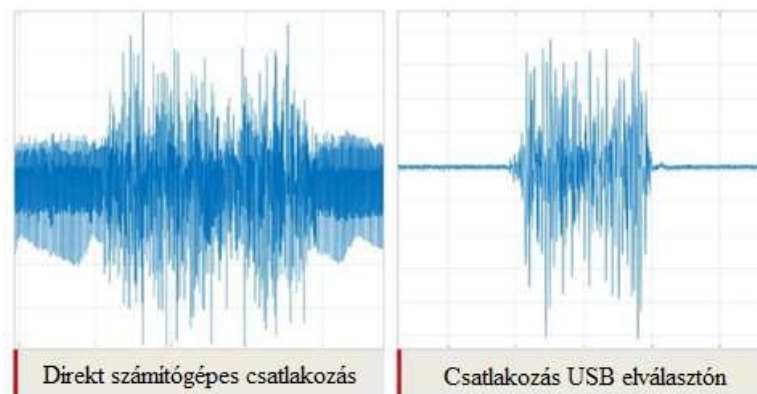
Ha az adatok rögzítése közben végezzük a szűrést, ez valós idejű feldolgozás során segít a zajok és „műtermékek-(artifacts)” valós idejű csökkentésében, ami javíthatja az adatminőséget. Továbbá a szűrés alkalmazása csökkentheti a zajok mennyiségét, így kezelhetőbbé és relevánsabbá téve az adathalmazt a betanításához, tehát az adatmennyiség csökkenését is elérhetjük. Ha a szűrés hardveresen megoldható például egy arduino vagy más mikrokontroller segítségével, akkor ez tehermentesítheti a fő számítógépet, és kevesebb lesz a valós idejű számítási igény. Ha a szűrést az adatok rögzítése közben végezzük, akkor olyan adatokat eredményezhet a szűrés, amelyek már előfeldolgozottak. Kevesebb utófeldolgozás szükséges modell betanítása során. Ha a szűrőket nem megfelelően választjuk meg,

előfordulhat, hogy fontos jellemzőket távolítunk el, vagy torzításokat vezetünk be az adatainkba.

Ha pedig a nyers adatok rögzítése utáni szűrést alkalmazunk, az nagyobb rugalmasságot biztosít, különböző szűrési technikákat és paramétereket lehet kipróbálni anélkül, hogy az eredeti adatokat megváltoztatnánk. Ha minden adatra ugyanazt a szűrőt alkalmazzuk, az biztosítja, hogy a betanító és teszt adatok egyenletesen legyenek feldolgozva. Ez kulcsfontosságú a modell teljesítményének konzisztenciája szempontjából. Továbbá lehetővé teszi, hogy különböző szűrési technikákat próbáljunk ki, és kiválasszuk a megfelelő szűrést a modell teljesítményükhöz.

Mind a két opciót kihasználtam az adatgyűjtéseim során, ezzel biztosítva, hogy jobb minőségű adatokat kapjak.

A valós idejű szűrést a fentebb említett USB izolátor biztosította, ami kiszűrte az elektromos zajokat és interferenciákat, amely magasfrekvenciás zajként megjelenhet a mérésekben azért, mert a mikrovezérlőt a személyi számítógépből közvetlen vezéreltem. Az izolátor által biztosított zajszűrés szemmel látható a következő ábrán (34. ábra).



34. ábra: EMG-jel direkt és leválasztóval történő csatlakoztatása (www.elemyo.com, dátum nélk.)

Ezen kívül még az Elemyo szenzorba beépített sávszűrő segítségével is szűrtem a jeleket. Ez a szűrő 8 és 200 Hz közötti frekvenciatartományt enged csak át. A beépített szűrő alsó határa tehát egy kicsit alacsonyabb, mint 10, ezzel biztosítva, hogy minden izom aktivitásból származó jel átmenjen a rendszeren, míg a felső határ kiszűri azokat a jeleket, amelyek már nem származhatnak izommozgásból.

Miután rögzítettem az izmokból származó jeleket, további szűrőket is használtam, ugyan ezen szűréseket már nem a valós időben végeztem el, de még a tanító algoritmusok előtt.

A mért adatok további szűrése azért is volt előnyös mivel, így tudtam kísérletezni a szűrők beállításával még mielőtt az adathalmazomat a neurális hálók feldolgozták volna. A

szűrők beállítása Hz-ben történt, mivel igaz, hogy az EMG adatok mV-ban voltak, de a mért jeleink különböző frekvenciákból állnak össze. Ezek miatt kell a frekvenciákat filterezni, így kontrollálva azt, melyik tartományt engedjük át és melyiket nem. Egyszerűen fogalmazva: mV az EMG jel amplitúdójára utal (az elektromos aktivitás erőssége). A Hz az elektromos aktivitás frekvenciájára utal, azaz arra, hogy milyen gyakran fordulnak elő jelciklusok (csúcsok és mélypontok) másodpercenként.

Többféle szűrőt is kipróbáltam, mire kiválasztottam a számomra megfelelőket, végül másodrendű Butterworth és Mozgóátlag (MA) szűrők alkalmazása mellett döntöttem, melyek előnyei és tulajdonságai kifejezetten az EMG adatok előfeldolgozásához szükséges kritériumoknak felelnek meg.

A másodrendű aluláteresztő Butterworth szűrő, elsősorban arra szolgált, hogy a magas frekvenciájú zajokat csökkentse, amik mérés közben a digitális eszközökből vagy a környezeti forrásból származhattak. Továbbá azért is volt előnyös, mivel ezen szűrő tulajdonsága, hogy az áteresztőszávjában a frekvenciaválasz lapos, vagyis nincs oszcilláció. Ennek köszönhető, hogy a jel hasznos komponenseit nem torzítja, csupán a zajos, magas frekvenciás részeket vágja le. A másodrendű megnevezés pedig arra utal, hogy a szűrő két pólussal rendelkezik, amely gyorsabb csillapítást biztosít a vágási frekvencián túl. A szűrő vágási frekvenciáját pedig 80 Hz-ra választottam, hogy még jobban megszabaduljak a nagyfrekvenciás zajoktól.

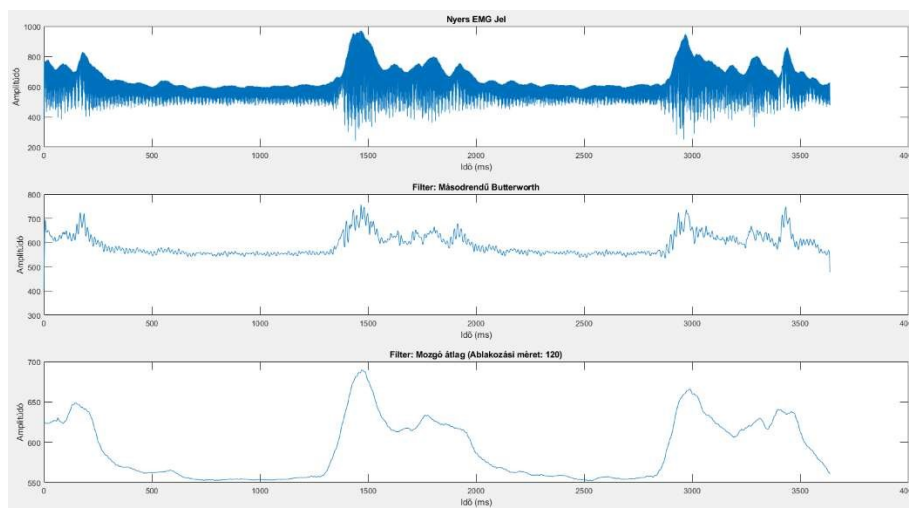
A mozgó átlag szűrés egy egyszerű, de hatékony simító szűrő, amely egy előre meghatározott ablakméret alapján az egyes minták átlagát számítja ki. Ez az ablak folyamatosan halad végig az adatokon, minden új mintával frissítve az átlagon. Majd a simítás következik, ami a jel ingadozását kezeli le, csökkenti a zajt, és a jel lassú változásait hangsúlyozza. Ez különösen fontos az elektromiográfiás jeleknél, mivel gyorsan változnak, és ezen jelösszetevők zajt jelenthetnek.

Az utóbbi szűrőnél az ablakválasztás mérete kritikus szerepű, ezért annak az értékét kellett jól megválasztani. A túl kicsi ablakméret ugyanis gyorsan változó értékeket eredményezhet, amelyek már nehezebben értelmezhetőek, míg a túl nagy ablakméret kisimíthatja az aktivitási csúcsokat. Én 120-at választottam ablakméretnek mind a két szűrőhöz.

A szűrők választása során egyéb szűrőket is kipróbáltam, így olyanokat is, amik például nemigen voltak alkalmasak a kívánt feladatra. Egy ilyen nem ajánlott szűrő izomvizsgálatokhoz például a Notch szűrő. Ezt gyakran alkalmazzák 50 Hz-es hálózati zajok eltávolítására, de az EMG jelek esetében nem a legjobb választás. Ezzel a szűrővel ugyanis nem csak a zajt, de a hasznos információkat is kiszűrhetjük az adataink közül, továbbá éles frekvencia-kivágást hajt

végre, ami a spektrumok torzulásához vezethet. A spektrumok torzulása pedig rontja a jelek interpretációját.

Végül sok kísérletezés után az általam választott két szűrőt (Butterworth, mozgó átlag) külön-külön használtam (35. ábra) az adatokon ezzel kipróbálva azt, hogyan javítják a jelek minőségét és feldolgozhatóságát.



35. ábra: Szűrők alkalmazása nyers elektromiográfiás jelen

Miután megtaláltam a megfelelő szűrőbeállításokat, már ténylegesen tudtam alkalmazni azokat a mért adataimon. Ezt már VSC-ben végeztem Python kód segítségével.

A szűrések megvalósítására két scriptet hoztam létre, az egyik egyesével alkalmazza a filtereket, a második pedig egymás után, de a működési alapelvük ugyan az. Elsősorban fontos könyvtárakat kell importálni, nekem a `scipy.signal` és a `numpy` volt elengedhetetlen a matematikai műveletekhez és a digitális szűréshez. Ez után mind a kettő szűrőnek (Butterworth, Mozgó átlag) létre kellett hozni egy saját függvényt, amiben a paramétereit lehet állítani. Mivel közös paramétereket alkalmaztam (vágási frekvencia, ablakozási méret), ezeknek egy változót hoztam létre, és itt adtam értéket hozzájuk (80Hz és 120 a MATLAB-ban kipróbáltak alapján). Végül még fontos beállítani a mintavételezési frekvenciát, ami megegyezett a méréskor beállított frekvenciával.

Ezek után a fájlok beolvasása következett, amik a mért adatokat tartalmazták Excel formátumban. Az első oszlop tartalmazta az mV értékét az izomnak, a második oszlop pedig az időt ms-ben. Ezek után minden fájlra lefutottak a szűrő függvények, és az adott szűrő nevét használva postfix-ként lementette a program az új fájlokat. A másik script pedig hasonlóan működik, annyi eltérés van, hogy egy új fájlt képez, úgy, hogy az már magában foglalja mind a két szűrő módosítását.

2.3. Mesterséges neurális hálózat megvalósítása

Dolgozatom során három gépi tanulási módszert valósítottam meg azért, hogy össze tudjam hasonlítani a teljesítményüket és tulajdonságaikat egymással, különböző szűrési technikák alkalmazása mellett. Az algoritmusok vizsgálata során figyelmet fordítottam a pontosság, precizitás, érzékenység és F1-score mutatók alapján történő összehasonlításra, hogy átfogó képet kapjak az egyes módszerek hatékonyságáról.

2.3.1. CNN

A CNN egy összetett mélytanulási hálózat, amely kifejezetten hatékony a mintázatok felismerésére, ezért előszeretettel használják képi adatok kategorizálására. Ugyanakkor nagyon jól teljesít időbeli adatok feldolgozása során is, és mivel az EMG adatsorunk időbeli jelek összessége, ezért próbáltam ki ezt az algoritmust.

Az általam implementált kód három fő szakaszra bontható: adatbetöltés és előfeldolgozás, CNN modell létrehozása, valamint a modell tanítása és tesztelése. Ezek a szakaszok alkotják a teljes folyamatot az adatok előkészítésétől a modell kiértékeléséig.

Azért is választottam a Python-t, mert nagyon felhasználóbarát, és a sok felhasználó miatt nagy mennyiségű előre elkészített könyvtár található, amiket fel lehet használni programok készítése során. A CNN algoritmus során például elengedhetetlen volt a keras könyvtár, ami a mélytanulási modell felépítésért felel, ez kezeli a különféle neurális hálózati rétegeket és az optimalizálási folyamatokat.

Az adatok betöltésért és előfeldolgozásáért felelős függvény (*load_data_from_folders*) az EMG adatsort tartalmazó Excel fájlokból tölti be az adatokat. Csak azokat a táblázatokat tölti be, amelyek a megfelelően tartalmazzák az adatokat (EMG érték [mV] és idő [ms]). Ha az adatsor hossza rövidebb, mint az adathalmaz maximális hossza, egy beépített függvény (*np.pad*) segítségével nullákkal egészíti ki a jelet a megadott hosszra, ez a padding. Ez azért fontos, hogy minden adat megfelelő hosszúságú legyen, és így hiba nélkül lehessen az adatokat a konvolúciós hálónak betáplálni (Haryanto, Sitanggang, Agmalero, & Rulaningtyas, 2020). Végül az egyes fájlok neve alapján az izomcsoportokat a program azonosítja. Ezen címkék felhasználásával a hálózat kategóriákat tanul meg osztályozni.

Az adatok betöltése után normalizálás és kategorizálás következik, amit a *run_cnn* függvény valósít meg. A normalizálásra azért van szükség, hogy a CNN könnyebben tanulhassa meg az adatok közötti különbségeket. Ehhez a *MinMaxScaler* függvényt használom, amely a jelet 0 és 1 közé skálázza. Az 1D konvolúciós réteg alkalmazása azért indokolt, mert az EMG adatok egy dimenziós időfüggő jelek, ahol a jel értékei időben követik egymást. Így a neurális

háló a jelsorozat mintázatait időbeli szekvenciák mentén tanulja meg, ezáltal képes lesz hatékonyabban felismerni a jelek jellemzőit anélkül, hogy felesleges komplexitást adnánk a rendszerünkhöz. Végül az egyes izomcsoportokat, vagyis a címkéket kategóriákká alakítjuk, amelyeket a CNN softmax rétege kezel. A *to_categorical* függvény segítségével minden kategóriához egy-egy bináris vektor tartozik, ami a hálózat számára egyszerűsíti a kimenet értelmezését.

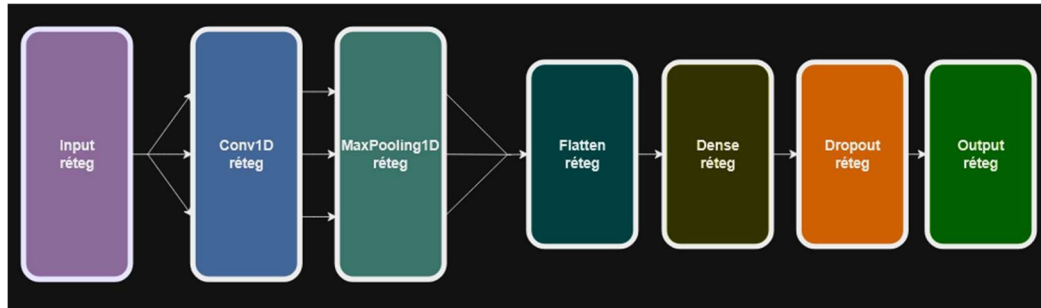
Ez után jön a CNN modell felépítése (*create_cnn_model*), függvény, ami a hálózatunk architektúráját hozza létre. Egy konvolúciós réteget használok, amely 64 szűrőt használ. Ez a réteg megtanulja felismerni az EMG jelben rejlő mintázatokat. A konvolúciós réteg után egy pooling réteg következik (MaxPooling1D), amely a konvolúciós réteg által generált jellemzők számát a felére csökkenti. Ezáltal a modell általánosítja az adatokat, ami segít a gyorsabb és hatékonyabb tanulásban. A flatten réteg laposítja a kimenetet, hogy egy 1D-s vektort hozzon létre. A laposítás annyit jelent, hogy a flatten réteg átalakítja az adatokat egy olyan sorozattá, ahol minden érték a konvolúciós műveletek során kinyert jellemzők egy-egy része, de ezek egy sorban helyezkednek el. Ezután egy Dense réteg (teljesen összekapcsolt réteg), 64 neuronnal, további jellemzőket tanul meg, amelyek magasabb szintű összefüggéseket képviselnek a konvolúciós réteg által létrehozott jellemzőkből. Ez a réteg arra összpontosít, hogy a korábbi réteg által kinyert információkból magasabb szintű kapcsolatokat hozzon létre, ezáltal segítve a végső osztályozást.

Majd következik a dropout réteg, ami segít a túlillesztés csökkentésében. Ez a réteg véletlenszerűen elhagyja a neuronok egy részét minden egyes tanulási lépésnél, így a hálózat nem fog túlságosan egy-egy jellemzőre összpontosítani.

Az utolsó pedig a kimeneti réteg, ami a *softmax* aktivációs függvényű réteg, ez felel az osztályozásért, ahol a neuronok száma megegyezik a lehetséges kategóriák (izomcsoportok) számával. A *softmax* függvény minden egyes kategóriára valószínűséget ad meg, így a modell döntést hozhat az osztályok között.

Miután a fenti kódrészleteket megalkottam, következhetett a modell tanítása és tesztelése (*run_cnn* függvény). Az adataimat felosztottam 80% tanító és 20% teszt halmazba a *train_test_split* segítségével. A tanításra 40 epochon keresztül került sor, 32-es batch méret mellett. Ez azt jelenti, hogy a tanulási folyamat 40 ismétlésből áll, tehát minden adatot 40 alkalommal lát és tanul belőle. A batch méret pedig azt jelenti, hogy a modell egyszerre 32 adatmintát dolgoz fel, majd frissíti a tanulási paramétereket, így a teljes folyamat több kisebb lépésből tevődik össze. A túlillesztés elkerülése végett, a tanító halmaz 20%-át a validációra

használtam fel (*validation_split*). A teszhalmazmal való kiértékeléshez előrejelzéseket kellett készíteni a tanult modellen. Az *accuracy_score* és a *classification_report* segítségével a modell pontosságát és osztályozási teljesítményét mértem minden izomcsoportra. Ezek az értékek mutatják meg, hogy a hálózat mennyire pontos az egyes izomcsoportok felismerésében.



36. ábra: Implementált CNN architektúra

Elégedett vagyok az általam készített CNN modell (36. ábra) eredményeivel, mivel így is sikerült vele elfogadható eredményeket elérni, amelyek jól tükrözik a modell hatékonyságát az általam végzett feladaton.

2.3.2. KNN (*K – Nearest Neighbors*)

Az algoritmus működése azon alapul, hogy egy új feljegyzést (rögzített adatot) összehasonlít a tanító adattal és megkeresi azokat, amelyek hasonlóak hozzá. Ebben az algoritmusban a „legközelebbi” fogalom egy távolságmétrikával van megadva, például euklideszi távolság alapján határozzák meg. Két pont n -dimenziós tere közötti euklideszi távolságot az alábbi képletek határozzák meg:

$$x_1 = (x_{11}, x_{12}, \dots, x_{1n}), x_2 = (x_{21}, x_{22}, \dots, x_{2n}) \quad (11)$$

$$dist(x_1, x_2) = \sqrt{\sum_{i=1}^n (x_{1i} - x_{2i})^2} \quad (12)$$

Ahol az x_1 és x_2 két rögzített érték, n attribútummal. A K -legközelebbi szomszéd osztályozó egy felügyelt tanulási algoritmus, ahol az új lekérdezés eredményét a KNN kategóriájának többsége alapján osztályozzák. A tanító minták n -dimenziós numerikus attribútumokkal vannak leírva, és minden minta egy pontot képvisel az n -dimenziós mintatérben. Ily módon az összes tanító minta az n -dimenziós mintatérben van eltárolva.

Az algoritmus működése nagyon egyszerű: úgy működik, hogy az adott lekérdezés példányához a legközelebbi tanító mintákat a minimális távolság alapján határozza meg. Miután összegyűjti a legközelebbi szomszédot, ezek többségi döntése alapján adjuk meg a lekérdezés példánynak előrejelzését. Az algoritmus adatai többváltozós attribútumokat (X_i) tartalmaznak, amelyek az objektum (Y) osztályozására szolgálnak. Tegyük fel, hogy a K értéke 8 (vagyis 8 legközelebbi szomszéd van) az algoritmus paramétereként. Ezután kiszámítja a lekérdezés

példány és az összes tanító minta közötti távolságot. Mivel csak kvantitatív X_i -k vannak, a következő lépésben pedig megtalálja a legközelebbi szomszédot. Minden tanító minta legközelebbi szomszédnak számít, ha a távolsága a lekérdezéshez kisebb vagy egyenlő, mint a K -edik legkisebb távolság. Más szóval, az összes tanító minta távolsága a lekérdezéstől sorba rendeződik és meghatározzák a K -adik minimális távolságot. Az ismeretlen mintát a K legközelebbi szomszéd közötti leggyakoribb osztályba sorolják. Ezek a tanítóminták az ismeretlen minta legközelebbi K szomszédjai, ahol a közelség az euklideszi távolság alapján van meghatározva. Tekintsünk egy olyan mintahalmazt $X = \{x_1, \dots, x_N\} \subseteq \mathbb{R}^p$, ahol az ismert osztályozások vannak tárolva, és minden egy adott osztályba tartozik $Y = \{w_1, w_2, \dots, 2s\}$. A legközelebbi szomszéd osztályozási szabály szerint egy ismeretlen osztályozású mintát a z osztályba sorolunk a legközelebbi szomszédja alapján, ahol $x_i \in X$ a legközelebbi szomszédja z -nek ha:

$$D(x_i, z) = \min\{D(x_l, z) | l = 1, 2, \dots, N\} \quad (13)$$

Itt D az euklideszi távolság két minta között az $\mathbb{R}^p$ térben. Ez a módszer az 1-NN szabály, mivel csak egyetlen szomszéd alapján osztályozza a mintát. A K -NN szabály pedig figyelembe veszi z -nek a K legközelebbi szomszédját és a többségi szabályt alkalmazza. Legyen t_l , $l = 1, 2, \dots, s$, a szomszédok száma az l osztályból z K -legközelebbi szomszédjai között,

$$\sum_{l=1}^s t_l = K \quad (14)$$

Ekkor z -t a j osztályhoz rendelhetjük, ha

$$t_j = \text{Max} \{t_l\} \quad (15)$$

Tehát lépésről lépésre hogyan számolható ki a K -legközelebbi szomszédok (KNN) algoritmus:

1. K paraméter meghatározása, amely a legközelebbi szomszédok számát jelenti.
2. Távolság számítása a lekérdezett példány és az összes tréning minta között.
3. Távolságok elrendezése és legközelebbi szomszédok meghatározása a K -edik legkisebb távolság alapján.
4. Legközelebbi szomszédok kategóriájának összegyűjtése (Y)
5. Többségi szavazással a legközelebbi szomszédok kategóriájának kiválasztása és ennek felhasználása a lekérdezett példány meghatározására

(Mohammed, Abduladhem, & Abbas, 2010)

Tehát a KNN algoritmus előnyei közé tartozik, hogy könnyen érthető és implementálható, gyorsan tanítható, valamint robusztus a zajos tanulóadatokkal szemben. Emellett képes bármennyi osztályba sorolni az adatokat, és jól működik nem-lineárisan

szeparálható adatok esetén is, ami megkülönbözteti más algoritmusoktól. A K értékének kiválasztása azonban kulcsfontosságú, mivel az algoritmus teljesítménye nagyban függ ettől. Az optimális K megtalálása időigényes lehet, de ha ez megvan, a tanítási idő általában alacsony. Hátrányai közé tartozik a magas számítási költség, amely az adatok és osztályok számának növekedésével arányosan emelkedik, a gyenge futási teljesítmény nagy adathalmazok esetén, valamint érzékenysége az adatok minőségére és irreleváns jellemzőkre. (Lopez-Bernal, Balderas, Ponce, & Molina, 2021)

A KNN egy viszonylag nagyon könnyen használható volt számomra és jól teljesített a próbák során. Természetesen itt is szükséges volt használni az ehhez tartozó fő könyvtárakat, amelyek már előre el vannak készítve a Python környezethez ilyen feladatokra (például `KNeighborsClassifier`, `cross_val_score`).

A könyvtárak importálása után pedig az adatok betöltése következett az adatokat tartozó mappákból, ez egy egyszerű függvény volt (*load_data_from_folder*), ami a mappában található Excel fájlokot iterált végig. A fájlnevek alapján ismeri fel az izomcsoportokat, amiket címkékként tárol. A fájlok tartalmát ezután listába gyűjti, majd, ha szükséges, adatbővítést végez, hogy minden minta méret konzisztens legyen, ez az úgynevezett padding (tömítés).

Itt a CNN-ben használt adatbeolvasó függvényt szintén fel tudtam használni, és ugyanazt valósítja meg itt is, mint az előző algoritmusban, az adatokat betölti és ellenőrzi.

Ezután egy jellemző kivonó függvény található (*extract_features*), ami az EMG adatokból jellemzőket vont ki, amit azért nagyon fontos kiemelni mivel a KNN nem idősorokkal dolgozik, hanem jellemzőkkel. Az általam kinyert jellemzők az átlag (a jel általános aktivitási szintjének mutatója), a szórás (amely az aktivitás ingadozásait méri), a maximum (a jel legnagyobb amplitúdója), a csúcs-értéktartomány (PTP) (a maximális és minimális érték közötti különbség, amely az aktivitási szélsőségeket mutatja), valamint az energia (az izom aktivitásának összegzett intenzitását tükröző mutató). Ezek a jellemzők kulcsfontosságúak voltak az osztályozás során, mivel az EMG idősorok bonyolultságát egyszerűbb, az osztályozás szempontjából releváns és könnyebben feldolgozható formába alakították.

Végül a main függvény következett, ami magát a fő feldolgozást és az osztályozást végezte. Ez a függvény a fenti két függvényt hívja meg egymás után sorban, úgy, hogy az adatok betöltése után egyből kinyerjük a jellemzőket. Ez után a kapott jellemzőket normalizálja, hogy az adatok értékei egy egységes skálára kerüljenek, ezzel javítva a modell teljesítményét. Ezután osztja szét az adatokat 70%-os tréning és 30%-os teszt arányban. A KNN modellt még

szükséges volt finomhangolni, hogy a lehető legjobb eredményt érjük el vele, ehhez bevezettem egy GridSearchCV nevű paramétert, ami keresést végez a KNN modell legjobb k értékének megtalálására (k a szomszédok száma, amit figyelembe vesz az osztályozásnál). A GridSearchCV módszer automatikusan teszteli a megadott paraméterértékeket (jelen esetben a k értékét) annak érdekében, hogy megtalálja azt a kombinációt, amely a legjobb teljesítményt nyújtja a modellen. Ezen kívül, a GridSearchCV a keresés során keresztvalidációt (esetemben 5szöröset) is végez, így megbízhatóbb eredményt biztosít a modell teljesítményéről különböző paraméterek esetén. Miután a script megtalálja a legjobb modellt, kiértékeli azt a teszt adatok segítségével. Ezt a metrikát pedig a *classification* report végzi, ami a pontosságot, precizitást és visszahívást jeleníti meg.

Végül keresztinvalidációt is végez a program, ami segít elkerülni az overfitting-et (túlilleszkedés). Ez például akkor történhet meg, ha a modell túl jól tanulja meg a tréning adatok mintázatait, de nem teljesít jól az új, ismeretlen tesztadatokon. A keresztinvalidáció segít megakadályozni ezt, mivel az adatok több különböző részén próbálja ki a modellt. Így van az biztosítva, hogy a modell nem csak a tréning adatokra lesz „behangolva”, hanem jobban tud általános ismeretlen adatokból is dolgozni.

Összeségében meg vagyok elégedve az általam készített KNN-nel.

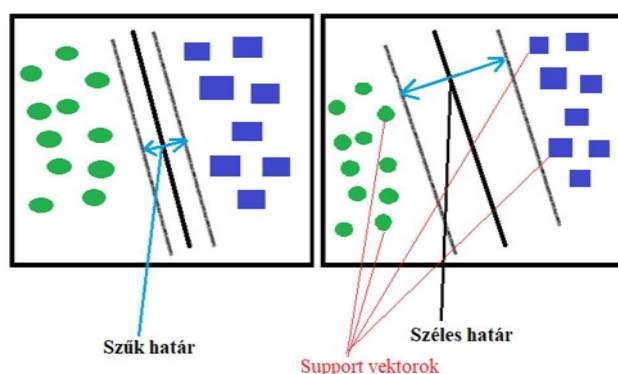
2.3.3. SVM (Support Vector Machine)

Az SVM-ek két kutatási terület metszéspontjába tartoznak: a Kernel módszerek és a nagy marginú osztályozók. Az SVM-et alkalmazzák jellemzők kiválasztására, idősor elemzésre, kaotikus rendszerek rekonstrukciójára, valamint nemlineáris főkomponens elemzésére. Az SVM-eket és a kapcsolódó módszereket egyre inkább használják a valós adatabányászatban is. A Support Vector Machine segítségével lineárisan elválasztható és nem lineárisan elkülöníthető adatokat osztályozhatunk.

Lineáris osztályozás:

Mielőtt az N-dimenziós hipersíkokat vizsgálnánk meg, nézzük meg először egy egyszerű 2-dimenziós példát. Tegyük fel, hogy egy osztályozást szeretnénk végrehajtani, és az adataink célváltozója két kategóriával rendelkezik. Feltételezzük továbbá, hogy két folytonos értékű prediktorváltozónk van. Ha ábrázoljuk az adatpontokat úgy, hogy az egyik prediktor értéket az X tengelyre, a másikat pedig az Y tengelyre helyezzük, egy olyan képhez juthatunk, mint amit az alábbi ábra szemléltet (37. ábra). Az egyik kategóriát téglalapokkal, a másikat oválisokkal jelöljük.

Ebben az idealizált példában az egyik kategóriához tartozó esetek a bal alsó sarokban, míg a másik kategória esetei a jobb felső sarokban helyezkednek el; az esetek teljesen elkülönülnek. Az SVM- elemzés megpróbál egy 1-dimenziós hipersíkot (azaz egy egyenest) találni, amely az eseteket a célkategóriáik alapján választja szét. Végtelen számú lehetséges egyenes létezik; az ábrán (37. ábra) két lehetséges egyenes látható. A kérdés az, hogy melyik egyenes a jobb és hogyan határozzuk meg az optimális egyenest. Az elválasztó vonallal párhuzamosan rajzolt halványabb vonalak jelzik a távolságot az elválasztó vonal és a legközelebbi vektorok között. A halvány vonalak közötti távolságot margin-nak (határnak) nevezzük. Azok a vektorok, amelyek a határ szélességét korlátozzák, a szupport vektorok.

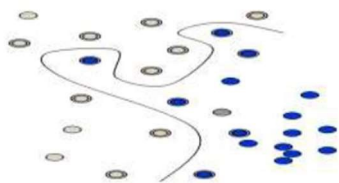


37. ábra: Kétdimenziós osztályozás (Himani & Mahesh H., 2012)

Az SVM elemzés megtalálja azt az egyenest (vagy általánosan hipersíkot), amely úgy van elrendezve, hogy a tartóvektorok közötti margó maximális legyen. Az előző ábrán a jobb oldali panelen látható egyenes jobb, mint a bal oldali panelen lévő. Ha minden elemzés két kategóriájú célváltozóból és két prediktorváltozóból állna, és az adatpontok egy egyenessel elválaszthatóak lennének, az nagyon egyszerű lenne. Sajnos ez általánosan nem így van, ezért az SVM-nek a következő problémákkal kell szembenéznie: (a) kettőnél több prediktorváltozóval, (b) az adatpontok nem-lineáris görbékkel történő szétválasztásával, (c) olyan esetekkel, amikor a klaszterek nem választhatók el teljesen és (d) kettőnél több kategóriás osztályozással.

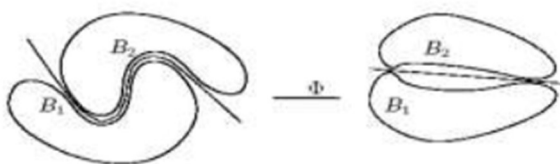
Nem lineáris osztályozás:

A legegyszerűbb módja két csoport elválasztásának egy egyenes vonallal, síkfelülettel vagy egy N-dimenziós hipersíkkal történik. De mi van akkor, ha a pontokat egy nem-lineáris régió választja el egymástól, mint az alábbi ábrán (38. ábra) látható:



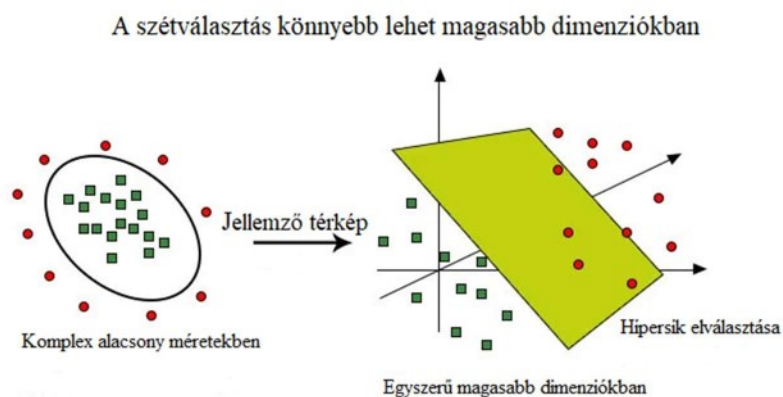
38. ábra: Multidimenziós osztályozás (Himani & Mahesh H., 2012)

Ebben az esetben egy nem-lineáris elválasztó vonalra van szükség. Ahelyett, hogy nem-lineáris görbéket illesztenénk az adatok mellé, az SVM ezt egy kernel függvény használatával oldja meg, amely az adatokat egy másik térbe képezi át, ahol egy hiperplán használható az elválasztáshoz (39. ábra).



39. ábra: Kernel funkcionális leképezése (Himani & Mahesh H., 2012)

A Kernel függvény átalakíthatja az adatokat egy magasabb dimenziós térbe, hogy lehetővé tegye az elválasztást (40. ábra).



40. ábra: Magasabb dimenziós leképezés (www.educba.com, dátum nélk.)

A Kernel függvény átalakítja az adatokat egy magasabb dimenziós térbe, ahol egy hipersíkkal elvégezhető a szeparáció. A Kernel függvények olyan algoritmusok egy osztályát képviselik, amelyek mintafelismerésre vagy elemzésre szolgálnak, és legismertebb elemük a Support Vector Machine. A tréning vektorok, x_i , Φ függvénnyel egy magasabb (akár végtelen)

dimenziós térbe vannak leképezve. Ezután az SVM egy lineáris elválasztó hipersíkot talál, amely a maximális határral rendelkezik ebben a magasabb dimenziós térben.

Az SVM-ben sokféle Kernel létezik, és a megfelelő kiválasztása a probléma jellegétől függ. A polinomiális Kernel változók kombinációit modellezi, az RBF (Radiális basis function) pedig köröket azonosít. Az ajánlott és leggyakrabban használt Kernel az RBF.

$$K(x_i, x_j) \equiv \Phi(x_i)^T \Phi(x_j) \quad (16)$$

Az SVM-nek is sajnos vannak korlátai. A legnagyobb kihívás a Kernel kiválasztása, amely egy adott problémához való legjobb illesztés megtalálása. Emellett a sebesség és a méret, különösen a nagy tanulási adathalmazok esetén, szintén korlátozás, mivel ilyenkor általában csak néhány támogatott vektort választanak, így csökkentve a teszteléshez szükséges számításigényt. Továbbá, a multiclass SVM osztályozók optimális tervezése is aktív kutatási terület. (Himani & Mahesh H., 2012)

Ez az algoritmus is az egyszerűbbek közé tartozik, ennek ellenére megfelelően lehet használni elektromiográfiás mérések kiértékelésére. A Python-on belül ennek is van saját könyvtára, amiket importálni kellett, mint például az SVC (Support Vector Classifier) és a StandardScaler könyvtár.

Ebben az algoritmusban is hasonlóan az előzőekhez egy adatbeolvasó függvénnyel kezdtem (*load_data_from_folder*), ami ellenőrzi az elvárt adathosszt és felismeri a felcímkézés alapján az adatainkat. Ha szükséges paddingot is alkalmaz, hogy minden idősorunk kompatibilis méretű legyen.

Az adatok betöltése után a jellemzőkivonó függvény (*extract_features*) lép működésbe, amely statisztikai jellemzőket von ki az EMG idősorokból. A kivont jellemzők megegyeznek a KNN-nél esetén kivont jellemzőkkel.

Ez után következik az SVM futtatására szolgáló függvény (*run_svm*). Először betölti az adatokat, és egy ellenőrzést végez, hogy ez sikeres volt-e. Majd az adatok felosztása következik tréning és teszt adatokra, viszont itt 80% és 20% arányban. Majd a már említett StandardScaler segítségével normalizálást hajt végre a függvény, mivel az adatok nagyságrendjére az SVM kifejezetten érzékeny.

Ezután következik a modell betanítása. A tanulási modell különböző Kernel típusokat támogat, amit én leimplementáltam, az a lineáris Kernel volt. A Kernel maga a függvény, amely lehetővé teszi az adatok magasabb dimenzióba történő transzformálását

A tanítás után a *run_svm* függvény előrejelzéseket készít a betanított modell segítségével a teszt adathalmazra, majd kiértékelést ad az *accuracy_score* és a *classification_report* segítségével.

Végül a main függvényünk az, ami bekéri a felhasználótól az adatok mappa útvonalát, és választást ad a Kernel típus kiválasztására, majd futtatja az SVM modellt.

Az EMG adatokon történő alkalmazása jó pontosságot mutatott.

3. EREDMÉNYEK

Elégedett vagyok az általam megvalósított adatgyűjtő és feldolgozó rendszerrel, amely az általam mért EMG adatok hatékony osztályozását teszi lehetővé a gépi tanulási algoritmusokkal kiegészítve. Sikerült egy olyan rendszert kialakítanom, amely képes többféle izomcsoport tevékenységét megkülönböztetni különféle szűrők és neurális hálók segítségével, otthoni körülmények között, viszonylag olcsó kivitelezéssel. Az alkalmazott algoritmusok (CNN, SVM, KNN) és a különböző szűrési eljárások (Butterworth, MA) kombinációjával változatos eredményeket kaptam, amelyeket az alábbiakban mutatok be.

3.1. Eredmények

Az alábbiakban láthatóak az egyes algoritmusok és szűrők kombinációinak eredményei. Az eredményeket pontosság, precizitás, recall (visszahívás) és F1-score mutatók alapján értékeltem, melyek segítségével egyértelműen megállapíthatók az algoritmusok közötti különbségek és teljesítménybeli eltérések. (1. táblázat) A metrikák macro átlagban vannak kiszámolva, így minden osztályhoz azonos súlyt adnak. (A macro átlag a klasszifikációs riportban egy olyan átlagolási módszer, amely az összes osztályra egyenlő súlyt ad az átlagolás során, függetlenül attól, hogy egyes osztályok milyen gyakorisággal fordulnak elő.)

A pontosság mutatja, hogy az algoritmus az összes tesztmintában milyen arányban adott helyes előrejelzést, és ez az általános teljesítménynek egy jó mutatója.

$$\text{Pontosság} = \frac{TP+TN}{TP+TN+FP+FN} \quad (17)$$

Ahol TP, TN, FP, és FN a különböző osztályozási kimeneteket jelölik (True Positive, True Negative, False Positive, False Negative).

A precizitás azt jelenti, hogy az adott osztályba sorolt elemek közül hány százalékot sorolt be helyesen az algoritmus, míg a recall megmutatja, hogy az adott osztályhoz tartozó összes elem közül hányat ismert fel helyesen az algoritmus.

$$\text{Precizitás} = \frac{TP}{TP+FP} \quad (18)$$

$$\text{Recall} = \frac{TP}{TP+FN} \quad (19)$$

Végül az F1-score jelenti a precizitás és a recall harmonikus átlagát, amely az algoritmus teljesítményének kiegyensúlyozott értékelésére szolgál. (Sokolova & Lapalme, 2009)

$$F1 = 2 * \left(\frac{\text{Precizitás} * \text{Recall}}{\text{Precizitás} + \text{Recall}} \right) \quad (20)$$

1. táblázat: Tanuló algoritmusok és szűrők kombinációinak kiértékelése

Algoritmus	Szűrő	Pontosság	Precizitás	Recall	F1-score
CNN	Butterworth	0,83	0,77	0,86	0,80
	MA	0,84	0,90	0,86	0,81
SVM	Butterworth	0,76	0,84	0,76	0,76
	MA	0,67	0,77	0,71	0,69
KNN	Butterworth	0,76	0,75	0,75	0,75
	MA	0,77	0,79	0,77	0,76

Az eredmények elemzése alapján látható, hogy a gépi tanulási algoritmusok és a szűrési módszerek kombinációi jelentős eltérést mutatnak a teljesítményben. Ez az eltérés az algoritmusok sajátosságaiból, a szűrési technikák tulajdonságaiból és az EMG jelek komplexitásából adódik.

A CNN algoritmus a legjobb teljesítményt a Mozgó Átlag (MA) szűrővel érte el, 0,84-es pontossággal, 0,90-es precizitással és 0,81-es F1-score értékkel. Az MA szűrő simítja a jelet, így a CNN számára optimális feldolgozási alapot biztosít, mivel kiemeli az alacsonyabb frekvenciájú komponenseket, ezzel csökkentve a zajt, ami a konvolúciós rétegek számára fontos a mintázatok hatékony felismeréséhez. A Butterworth szűrővel is kedvező eredményeket hozott, 0,83-as pontossággal és 0,80-as F1-score értékkel, ami az aluláteresztő szűrő hatékonyságára utal a magas frekvenciás zaj csökkentésében. Ez a konvolúciós rétegek számára fontos, mivel a jel integritásának megőrzése mellett lehetővé teszi az időbeli mintázatok felismerését, ami segíti a pontos osztályozást.

Az SVM algoritmus esetében a Butterworth szűrővel elért 0,76-os pontosság mutatja a legjobb teljesítményt, mivel az alacsony frekvenciájú komponensek megerősítése segíti az SVM döntési határvonalainak felépítését. Az MA szűrő hatására az SVM pontossága 0,67-ra csökkent, mivel a jel simítása miatt elvesznek azok az éles mintázatok, amelyek segíthetnének a hiper-sík alapú szeparációt. Az SVM jól reagál azokra a jelekre, amelyekben az osztályokat elválasztó éles határok jobban megmaradnak, amit a Butterworth szűrő eredményei is tükröznek.

A KNN algoritmus esetében az MA szűrővel elért 0,77-es pontosság és 0,79-es precizitás jelzi, hogy a KNN előnyben részesíti a simított jeleket. A KNN a szomszédos minták közötti távolságokat használja az osztályozás során, ezért a zajmentes, homogén jelstruktúrák javítják az osztályozás hatékonyságát. Ezzel szemben a Butterworth szűrővel kapott 0,76-os

pontosság is kedvező, de az összetettebb jelmintázatok megőrzése miatt az eredmény némileg gyengébb, mivel a KNN érzékenyebb a zajra és az extrém értékekre.

Összefoglalva megállapítható, hogy bár az eredmények nem érték el az általam olvasott tanulmányokban közölt legjobb pontossági szinteket például: (Bekir, 2014) , (Chaoming, Bowei, Yixuan, Jin, & Shuo, 2020), (Mokaya, és mtsai., 2023), így is megbízhatóak voltak. Bár fontos azt is megemlíteni, hogy nem találtam olyan forrást, ahol konkrétan az izomcsoportokat próbálták mozgások alapján elkülöníteni egymástól. Az eredmények szempontjából korlátozó tényező volt az, hogy a méréseket kizárólag magamon végeztem el, illetve egyetlen szenzort használtam az EMG jelek rögzítésére. Emellett az adathalmaz mérete is kisebb volt a közleményekben használt mintaszámhoz képest, ami befolyásolhatta az algoritmusok generalizációs képességét. Ezen korlátok ellenére a kapott eredmények kedvezőek, és azt mutatják, hogy a megvalósított rendszer képes az izomcsoportok közötti különbségtételre. Az eredmények diagrammos kiértékelése a mellékletekben megtekinthető (1. Melléklet).

3.2. Statisztikai elemzés

Fontosnak tartottam, hogy elemezzem az izomcsoportokat, mivel a dolgozatom során célzottan szerettem volna az egyes izmok közötti különbségeket azonosítani és az izomtevékenység specifikus mintázatait felfedezni.

Az ANOVA (Analysis of Variance) módszert választottam az elemzéshez, mivel ez egy olyan robosztus elemzés, amely lehetővé tette a többi izomcsoport közötti átlagos aktivitásértékek összehasonlítását, és segített kimutatni azt, ha szignifikáns eltérések voltak az izomcsoportok között. Az ANOVA-val az egyes izomcsoportok közötti varianciát vizsgáltam, így hasznos volt annak eldöntésében, hogy a próbaméréseim során az izomcsoportok közötti eltérések statisztikailag igazolhatók-e, vagy csak véletlen ingadozásoknak köszönhető az eltérés.

Az ANOVA statisztikai módszer célja a több, különböző csoport között fennálló átlagos különbségek szignifikanciájának meghatározása a varianciák elemzésén keresztül. Működési alapelve, hogy a vizsgálat a teljes varianciát két összetevőre bontja: csoporton belüli varianciára és csoportok közötti varianciára. Az ANOVA során a csoportok közötti varianciát a csoporton belüli varianciához viszonyítjuk, és egy F-próbát alkalmazunk, amely a két varianciarányados segítségével számszerűsíti, hogy a csoportok közötti különbségek nagyobbak-e a csoporton belüli varianciánál. Az eredmény egy F-statisztika, amelyhez egy p-érték tartozik; ha a p-érték kisebb, mint egy előre meghatározott szignifikancia szint (általában 0,05), akkor a csoportok

közötti különbség statisztikailag szignifikánsnak tekinthető. Tehát a számítás során először meghatározzuk az egyes csoportokon belüli varianciákat, majd kiszámítjuk a csoportok közötti varianciát. Az F-érték ezen két variancia viszonyának mértéke. A kapott F-értéket és a hozzá tartozó p-értéket a Fisher-féle F-eloszlás alapján értékeljük, amely kijelöli a szignifikancia szintet. (Wilcox, 2002)

Az ANOVA működési elve a teljes variancia két összetevőre való felbontása:

1. Csoportok közötti variancia ($SS_{between}$): Az eltérések a csoportátlagok és a teljes mintaátlag között.

$$SS_{between} = \sum_{j=1}^k n_j (\bar{x}_j - \bar{x})^2 \quad (21)$$

Ahol n_j a j-edik csoport mintaelemeinek száma, $\bar{x}_j$ a j-edik csoport átlaga, és $\bar{x}$ a teljes mintaátlag. Ez az eltérés a csoportok átlagai és a teljes mintaátlag között.

2. Csoporton belüli variancia (SS_{within}): Az eltérések az egyes csoporton belüli értékek és csoportátlag között.

$$SS_{within} = \sum_{j=1}^k \sum_{i=1}^{n_j} (x_{ij} - \bar{x}_j)^2 \quad (22)$$

Ahol x_{ij} az adott csoport egyedi értékei, és $\bar{x}_j$ az adott csoport átlaga.

3. A teljes variancia (SS_{total}): Az egyes megfigyelések és a teljes mintaátlag közötti eltérések négyzetösszegét jelenti.

$$SS_{total} = \sum_{i=1}^N (x_i - \bar{x})^2 \quad (23)$$

Ahol x_i az egyes megfigyeléseket jelenti, $\bar{x}$ a teljes mintaátlagot, és N a teljes minta méretet.

4. F-érték: A csoportok közötti és a csoportokon belüli varianciák hányadosa.

$$F = \frac{\frac{SS_{between}}{df_{between}}}{\frac{SS_{within}}{df_{within}}} \quad (24)$$

Ahol $df_{between} = k - 1$ (k a csoportok száma) és a $df_{within} = N - k$ (N az összes megfigyelések száma).

5. Négyzetes középérték (MS): Ez a variancia egyfajta mértéke, amelyet a SS_{total} és a hozzá tartozó szabadságfok (df) hányadosaként adunk meg:

$$MS = \frac{MS_{between}}{MS_{within}} \quad (25)$$

6. p-érték: Azt jelzi, hogy mekkora a valószínűsége annak, hogy a megfigyelt különbségek véletlenszerűek. (NIST/SEMATECH e-Handbook of Statistical Methods, dátum nélk.)

Az elemzést nyers próbaméréseken végeztem, mert arra voltam kíváncsi, hogy a feldolgozatlan adatokból indulva láthatóak-e az aktivitás alapvető különbségei. A nyers adatok elemzése azért volt előnyös, mert ezzel elkerültem a szűrési eljárások által okozott torzításokat vagy adatvesztéseket, és ezáltal megőriztem az izomműködés természetes variabilitását.

Az ANOVA teszt eredményei (2. táblázat) egyértelműen jelzik, hogy az izomcsoportok közötti EMG aktivitási értékek között statisztikailag szignifikáns különbségek vannak. Az elemzés során a varianciát két fő összetevőre bontottam: a csoportok közötti varianciára, amely az izomcsoportok átlagos EMG aktivitásának eltéréseit tükrözi, és a csoportokon belüli varianciára, amely az izomcsoportok mért adatainak szórását jelzi. A csoportok közötti szumma-négyzetek értéke jóval kisebb, mint a teljes variancia, de még így is jelentős hatással bír az F-statisztikára.

Az F-statisztika kiszámítása során magas F értéket kaptam. Ez az eredmény azt sugallja, hogy az izomcsoportok közötti eltérések nem véletlenszerűek, hanem jelentős különbségek mutatkoznak az izomcsoportok között. Az F-értékhez tartozó p-érték gyakorlatilag 0, ami a 0,05-ös szignifikanciaszinten kimagasló bizonyításként szolgál arra, hogy az izomcsoportok eltérő EMG aktivitási szinteket mutatnak.

2. táblázat: ANOVA – teszt eredmények

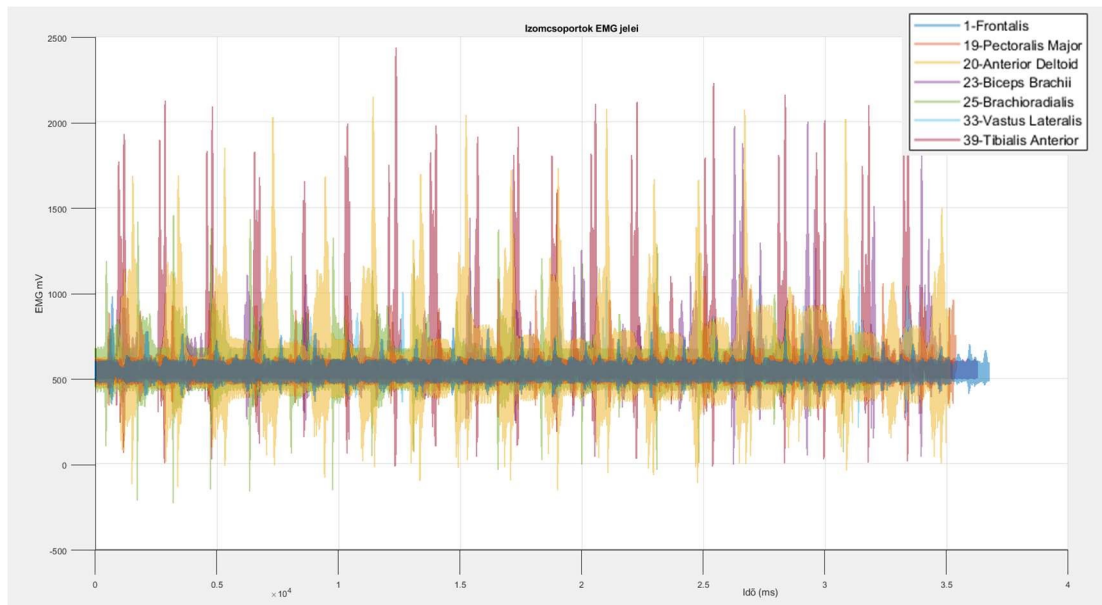
	Szumma-átlag	Szabadságfok	Négyzetes középérték	F-érték	p-érték
Csoportok	3.4089e+08	6	56815000	2208,3	0
Error	6.3217e+09	245711	25728	-	-
Összegzés	6.6626e+09	245717	-	-	-

A fenti eredmények igazolják, hogy a különböző izomcsoportok aktivitása között valódi eltérések találhatók, amelyek anatómiai és funkcionális különbségek különbségekből erednek.

A mellékletekben található ábra (2. Melléklet) az egyes izomcsoportok nyers EMG méréseit ábrázolja, amelyek szórása és eloszlása vizuálisan is megerősíti az ANOVA teszt során feltárt eltéréseket. Az x-tengelyen az egyes izomcsoportok nevei találhatók, míg az y-tengelyen az EMG értékeket ábrázoltam mV-ban. Az ábrán az látható, hogy minden izomcsoporthoz több adatpont tartozik, amelyek az adott izomcsoportok EMG aktivitásának szóródását tükrözik. Az izomcsoportonkénti különbségek vizuálisan is megfigyelhetők: az egyes csoportok mért aktivitási értékeinek eloszlása eltérő nagyságú és szórású, ami jól szemlélteti az előző teszt eredményeit.

A következő ábra (41. ábra, másik perspektíva: 3. Melléklet) pedig segít vizuálisan is elkülöníteni az EMG aktivitásoknak az időbeli mintázatát, egyes izomcsoportok esetében. Ez a

szemléltetés kiemeli a csoportosított mérések dinamikáját, és vizuális megközelítéssel támasztja alá az ANOVA által bebizonyított különbségeket. Az izomcsoportok időbeli aktivitásmintázatairól nyújt vizuális visszajelzést, nemcsak a mért amplitúdók skáláját, hanem az aktivitás karakterisztikáját is ábrázolja.



41. ábra: Izomcsoportok eltéréseinek szemléltetése próbaméréseken

Az ANOVA teszt futtatását, az adatok értékelését és kirajzolását MATLAB szoftver segítségével végeztem el, amely biztosította a szükséges statisztikai számításokat és az eredmények gyors feldolgozását.

3.3. KÖVETKEZTETÉSEK ÉS JAVASLATOK

A dolgozat során elért eredményeim alapján a különböző szűrőkkel és tanuló algoritmusokkal elért teljesítmény összehasonlítása azt mutatja, hogy a CNN modell kiemelkedően jól teljesített mind pontosság, mind F1-score szempontjából. Az ANOVA elemzés alapján pedig az izomcsoportok közötti EMG aktivitási különbségek szignifikánsak, ami azt jelenti, hogy az izomcsoportok különálló aktivitási mintázatokkal rendelkeznek. Ez a szignifikancia az algoritmusok teljesítményére is hatással volt, mivel az egyes izomcsoportok különböző mintázatait eltérő módon értékelte ki. A dolgozatom révén kimutatott eredmények megerősítik, hogy az izomcsoport-specifikus mintázatok lényeges tényezők az EMG jelek feldolgozásában, és a különböző szűrők és tanuló algoritmusok eltérően képesek kezelni ezeket a különbségeket.

Fontos megjegyezni, hogy bár a kapott eredmények kedvezőek és informatívak, a szakirodalomban dokumentált, esetenként orvosi mérőeszközökkel végzett kutatások gyakran magasabb pontossági értékeket érnek el. Ennek egyik oka, hogy a dolgozatban használt mérésekben, egyetlen szenzort használtam (Elemlyo 1.4), ami nem minősül orvosi eszköznek. Ennek köszönhetően nem tudtam olyan pontosan megejteni a méréseimet. Ezentúl az adathalmazom is korláatosnak tekinthető, mert egyetlen alanyra korlátozódott a mérés, így nem biztosította a legjobb adatbázist a legjobb kategorizálási eredmények elérésére és szélesebb körű következtetések levonására.

Fejlesztésként a jövőben javasolt a szenzor minőségének javítása, esetlegesen több szenzor használata egy mérés alkalmával, ezzel lehetőséget biztosítva a finomabb izomaktivitási jelek megbízhatóbb és pontosabb rögzítésére. A több szenzor még részletesebb információt biztosíthat az izom körüli izomaktivitás mintázatokról.

A jelfeldolgozó egység továbbfejlesztése során különösen érdemes a **valós** idejű feldolgozás lehetőségére összpontosítani. A betanított modellek közül az SVM és a CNN algoritmusok könnyedén optimalizálhatók fix adatstruktúrával MCU-ra, és a már betanított modellek különösen gyorsak, ami előnyös valós idejű alkalmazások esetén. A KNN modell esetében hatékony adatrepresentációra van szükség, hogy gyorsan meghatározható legyen az új pont helye, amihez léteznek hatékony algoritmusok. Emellett a digitális szűrők finomhangolása és a többfokozatú szűrés alkalmazása is hozzájárulhat a jelek tisztaságának javításához, és így a teljes algoritmus teljesítményéhez.

A mérési környezet és az adatgyűjtés is tovább fejleszthető, például lehetséges több alany bevonása, különböző életkorú csoportok és különböző fizikai állapotú személyek részvétele, amellyel nagyobb variabilitást lehetne vinni az adatbázisba, ezzel lehetővé téve az eredmények szélesebb körű általánosíthatóságát.

Érdekes lenne több tanuló algoritmus kipróbálása, a mesterséges intelligencián alapuló fejlettebb algoritmusok alkalmazása, például a mélytanulási modellek további finomhangolása, vagy új architektúrák, mint a rekurzív neurális hálózatok (RNN) vagy a konvolúciós hálózatok kombinált alkalmazása. Az ilyen algoritmusok képesek lehetnek pontosabb mintafelismerésre, és jobban adaptálódnak a komplex, nemlineáris mintázatokhoz, amelyek az EMG adatokban gyakran előfordulnak. Az ilyen módszerek implementálása és finomítása hozzájárulhat az EMG-alapú izomcsoport-elemzés jövőbeli alkalmazhatóságának és hatékonyságának növeléséhez.

4. ÖSSZEGZÉS

Munkám során az eltérő izomcsoportok aktivitásának feltérképezésével foglalkoztam EMG szenzorok és neurális hálózatok segítségével. Ehhez elkészítettem a szükséges adatgyűjtő rendszert és az adatok feldolgozásához szükséges szoftvereket. Hét izomcsoportból mérési EMG adatokat gyűjtöttem, amiket sikeresen használtam fel a tanuló algoritmusaim (CNN, SVM, KNN) betanítására, melyek a minták alapján elfogadhatóan teljesítettek az izomcsoport minták kategorizálásában.

A kapott eredmények alátámasztják, hogy az EMG alapú izomcsoport-elemzés megfelelő jelfeldolgozási technikákkal és gépi tanulási algoritmusokkal kiegészítve, alkalmas az izomcsoportok közötti aktivitási különbségek azonosítására és vizsgálatára. Az általam elért mérések hatékonysága fejleszthető más algoritmusok és pontosabb szűrési eljárásokkal, így gyakorlati alkalmazhatósága később javítható a munkámnak. Ebben a dolgozatban alkalmazott módszerek széles körben hasznosak lehetnek olyan iparágak számára, ahol fontos az izomaktivitás mérése és az izomcsoportok pontos megkülönböztetése. Az izomcsoportok mozgásainak elkülönítése hasznos lehet a robotizált protézisek alkalmazása, sportteljesítmények mérése, sportolók edzése során. Az EMG alapú elemzések lehetőséget adnak arra, hogy az izomaktivitás pontosabb és személyre szabottabb módon váljon mérhetővé, amely új perspektívát nyithat a rehabilitáció, sport, robotizált protézisek és az ergonómia területén.

Összegezve elmondhatom, hogy sikerült megvalósítanom egy adatgyűjtő rendszert ArduinoUno és egy Elemyo 1.4 EMG szenzor segítségével, majd ezt felhasználva hét izomcsoportból elfogadható minőségű EMG adathalmazt gyűjtenem. Ezen adatokat feldolgoztam különféle szűrési technikákkal (Butterworth, MA), hogy megfelelő minőségű legyen a tanuló algoritmusaim számára. Ezt követve három tanuló algoritmust (CNN, SVM, KNN) implementáltam, amelyeknél aztán megvizsgáltam, hogy milyen hatékonysággal tudják feldolgozni az adataimat.

Az izomcsoportok közötti aktivitási különbségek vizsgálata az élet számos területén kiemelt fontossággal bírhat, erre alkalmas lehet az általam leírt és kipróbált módszer, illetve később annak továbbfejlesztett változatai.