VILNIAUS GEDIMINO TECHNIKOS UNIVERSITETAS

FUNDAMENTINIŲ MOKSLŲ FAKULTETAS

CHEMIJOS IR BIOINŽINERIJOS KATEDRA

Taisija Dėmčėnko

BALTYMO – LIGANDO JUNGIMOSI *IN SILICO* VERTINIMO FUNKCIJOS KŪRIMAS NAUDOJANT MAŠININĮ MOKYMĄSI

CREATING MACHINE LEARNING BASED SCORING FUNCTION FOR PROTEIN – LIGAND BINDING *IN SILICO*

Baigiamasis bakalauro darbas

Bioinžinerijos studijų programa, valstybinis kodas 612J76001

Biotechnologijos studijų kryptis

Vilnius, 2020

VILNIAUS GEDIMINO TECHNIKOS UNIVERSITETAS

FUNDAMENTINIŲ MOKSLŲ FAKULTETAS

CHEMIJOS IR BIOINŽINERIJOS KATEDRA

TVIRTINU
Katedros vedėjas

_____
(Parašas)

_____
(Vardas, pavardė)

_____
(Data)

Taisija Dėmčėnko

BALTYMO – LIGANDO JUNGIMOSI *IN SILICO* VERTINIMO FUNKCIJOS KŪRIMAS NAUDOJANT MAŠININĮ MOKYMĄSI

CREATING MACHINE LEARNING BASED SCORING FUNCTION FOR PROTEIN – LIGAND BINDING *IN SILICO*

Baigiamasis bakalauro darbas

Bioinžinerijos studijų programa, valstybinis kodas 612J76001

Biotechnologijos studijų kryptis

Vadovas_____ _____ _____
(Pedag. vardas, vardas, pavardė) (Parašas) (Data)

Konsultantas_____ _____ _____
(Pedag. vardas, vardas, pavardė) (Parašas) (Data)

Konsultantas_____ _____ _____
(Pedag. vardas, vardas, pavardė) (Parašas) (Data)

Vilnius, 2020

Turinys

ĮVADAS

Baltymo sąveikos su kitomis molekulėmis supratimas yra svarbus skirtingose biochemijos srityse, ypač farmacijos ir biofarmacijos srityse.[6][20] Proteino–ligando kompleksų dėsnių suvokimas yra reikšmingas naujų vaistų atradimui - vaistai paprastai užima ligando vietą komplekse. Naujo vaisto kūrimas įprastai prasideda nuo didelės kolekcijos ligandų rinkimo; kiekvieną iš šių ligandų planuojama išanalizuoti *in vitro* kaip potencialų vaistą. Aišku, tai reikalauja daug resursų ir laiko. Baltymo–ligando komplekso modeliavimas *in silico* ir tolimesnė aktyvių ligandų virtuali atranka yra metodas, leidžiantis žymiai palengvinti naujo vaisto kūrimo procesą laiko ir kainos atžvilgiu. Šio metodu galima palyginus greitai išskirti molekules, turinčias didžiausią potencialą, vietoje pilnos kolekcijos analizės. [3][20]

Sparčiai didėjanti skaičiavimo galia leidžia naudoti vis sudėtingesnius algoritmus baltymų modeliavimui bei jo surišimo su ligandu prognozavimui. Ypač gerus rezultatus parodo mašininio mokymosi algoritmai, nes dažnai nereikalauja papildomų žinių apart baltymo ir ligando molekulinės struktūros ir/ar fiziko–cheminių savybių. Tačiau mašininio mokymosi algoritmo rezultatus priklauso nuo turimų duomenų kiekiu ir kokybe: kuo duomenys kokybiškesni, tuo būna geresni tokių modelių rezultatai. Baltymų ir kitų junginių struktūrų ir parametrų duombazės nuolat pildosi naujais arba patobulintais duomenimis. Todėl kuriami vis nauji mašininio mokymosi modeliai, apmokinti ant dar išaugusios ir patobūlintos baltyminių kompleksų duombazės.[14] Vienas toks modelis, paremtas dirbtiniu neuroniniu konvoliuciniu tinklu, bus pasiūlytas šitame darbe.

Šio darbo tikslas yra sukurti naują baltymo–ligando surišimo prognozavimo įrankį (vertinimo funkciją), pagrįstą mašininio mokymosi algoritmu. Šio darbo iškeliami uždaviniai:

1. Išanalizuoti jau sukurtas vertinimo funkcijas, kurios sprendžia tą patį problemą, ir išsirinkti vieną iš jų kaip atramos tašką.
2. Paruošti modelio kodą, išrinkti kokybišką baltymo–ligando kompleksų rinkinį ir apmokyti modelį naudojant šį rinkinį.
3. Įvertinti apmokyto modelio prognozavimo rezultatus ir palyginti su panašių modelių įvertinimais.

1. LITERATŪROS APŽVALGA IR ANALIZĖ

1.1. Baltymų sąveikos su skirtingomis molekulėmis

Baltymai yra didelės ir sudėtingos biologinės molekulės, atliekančios skirtingas funkcijas, tarp kurių yra katalizės funkcija. Tokie baltymai vadinami fermentais ir turi aktyvųjį centrą, prie kurio prisiriša kita molekulė; sudarytas kompleksas skatina arba slopina tam tikrą reakciją. Molekulė (ligandas) gali prisijungti vandenilinių ryšių, elektrostatinių ryšių, hidrofobinės sąveikos ar Van der Valso jėgų dėka.[5] Šis surišimas paprastai yra trumpalaikis bei grįžtamasis.

Veikiantys organizme baltymai būna tretinės arba ketvirtinės struktūros. Tretinė struktūra yra baltymo kompaktiškas susilankstymas erdvėje su skirtingų ryšių (disulfidinių, vandenilinių bei kitų) pagalba. Baltymo struktūrą galima nustatyti skirtingais metodais (rentgenostruktūrine analize, branduolių magnetinio rezonanso metodu ir t.t.),[5] ir nustatytos struktūros yra renkami į duombazes (pvz. PDB). Tačiau to neužtenka proteino–ligando sąryšio prognozavimui: reikia dar nustatyti aktyvaus centro vietą ir nustatyti, ar jis gali pririšti pasirinktą ligandą. Fermentai turi didelį specifiškumą - jie geba atrinkti tik vieną specifinę molekulę iš milijonų skirtingų substratų.

Aktyvusis centras paprastai užima tik 10–20 % baltymo masės, tačiau yra svarbiausia fermento dalis. Jis dažniausiai yra sudarytas iš 3–4 aminorūgščių, prie kurių jungsis ligandas.[5] Skirtingų fermentų atvejais tai gali būti mažos molekulės, nukleorūgštys arba kiti baltymai ir peptidai. Šiame darbe bus analizuojami tik baltymo ir mažos molekulės kompleksai.

Ankstesnė baltymo–ligando komplekso „spynos ir rakto" teorija teigė, kad ligandas privalo turėti tą patį dydį ir formą, kaip ir baltymo aktyvaus centro kišenė. Tačiau eksperimentiniu būdu buvo nustatyta, kad susiriša kompleksai, kurių molekulių formos nepilnai sutampa.[5] Todėl išsivystė naujesnė „indukuoto įtalpinimo" teorija, kuri teigia, kad priartėjęs ligandas keičia baltymo aktyvaus centro struktūrą, taip pat gali ir pats nežymiai pasikeisti. Tačiau net to neužtenka, nes abi teorijos neatsižvelgia į pačio baltymo dinamiškumą. Baltymai nuolat keičia savo konformaciją skirtingose aplinkose; tik būdami specifinėje konformacijoje, baltymas gali surišti specifinį ligandą. Taip teigia naujausia „konformacijos išrinkimo" teorija. Šios trys teorijos nepaneigia viena kitos - skirtingose atvejuose buvo eksperimentiškai nustatyti visų trijų mechanizmų surišimai.[5]

1.1.1. Baltymo su ligandu surišimo konstanta

Galima pavaizduoti baltymo–ligando komplekso sudarymą šia pusiausvyros lygtimi:

$$L + P \underset{K_{off}}{\overset{K_{on}}{\rightleftharpoons}} LP \tag{1}$$

kur L - ligandas, P - baltymas, LP - baltymo–ligando kompleksas, K_{on} ir K_{off} - surišimo ir disocijavimo reakcijų kinėtinės konstantos.[5]

Nors baltymo–ligando jungimosi prognozavimo algoritmai paprastai prognozuoja, ar vyks surišimas, prognozavimui naudojama ne surišimo konstanta K_b, o atvirkštinė jai disociacijos konstanta K_d. Ji priklauso nuo visų trijų koncentracijų:

$$K_d = \frac{1}{K_b} = \frac{[L][P]}{[LP]} = \frac{K_{off}}{K_{on}} \tag{2}$$

Pusiausvyros konstantos vienetai yra M (g/mol).

Zubrienė ir Matulis [13] aprašo skirtumą tarp stebimos ir tikros surišimo konstantos. Jie atkreipia dėmesį į tai, kad baltymas gali egzistuoti keliose konformacijose, tarp kurių tik viena gali susirišti su ligandu. Baltymo konformacijos pakeitimas gali reikalauti nemažai energijos, ir jei tirpale yra daugiau neaktyvios baltymo konformacijos, išmatuota K_d parodys tikros surišimo konstantos ir konformacijos keitimui reikalingos energijos sumą. Tikra surišimo konstanta bus aukštesnė. Taip pat autoriai pabrėžia, kad stebima K_d priklauso nuo tirpalo pH, tuo metu tikroji K_d nepriklauso. Tą reikia turėti omenyje renkant duomenis mašininio mokymosi modeliui, nes tai gali paveikti modelio prognozių tikslumą.

Kitas svarbus aspektas yra vandens molekulių koordinatės baltymo struktūroje. Straipsniuose [3] ir [18] autoriai kalba apie vandens ypatingą svarbumą proteino surišimui su ligandu. Proteinai *in vivo* yra apsupti vandens molekulėmis, kurios veikia baltymo konformaciją bei dalyvauja komplekso surišime. Straipsnio autoriai nurodo, kad vandeniui skiriama per mažai dėmesio, nes ji atlieka daug sudėtingų funkcijų, o dėl to dažnai sunku prognozuoti surišimo stiprumą, net kai yra nustatyta baltymo struktūra. Apie tai rašo ir Sethi et al.[20]; jie pabrėžia, kad informacijos apie vandens molekules yra per mažai dėl vandenilio koordinačių trukumo rentgeno-struktūriniu būdu nustatytose baltymų kristalinėse struktūrose, bei patikimų teoretinių žinių apie vandens ir ligando sąveiką nepakankamumo. Iš kitos pusės, Chen et al.[4] parodo, kad vandenio molekulių įvedimas į struktūras nepagerino aprašomo modelio prognozavimo tikslumo. Šiame darbe vandeniui nebus skirta dėmesio.

1.2. Ligando įvedimas į baltymą

Ligando įvedimas į baltymą (angl. *molecular docking*) yra molekulių projektavimo technika. Šios technikos esmė yra naudojant baltymo ir ligando erdvines struktūras, rasti tokias molekulių padėtis, kad susijungus kompleksui šiose padėtyse, ryšys būtų kuo stipresnis. Ligando įvedimo į baltymo technika sprendžia du uždavinius: ligando pozicijų aktyviajame centre prognozavimas, ir tos pozicijos surišimo su baltymu stiprumo įvertinimas.[1][2] Antra uždavinį atlieka vertinimo funkcija (angl. *scorinc function*).[3] Sethi et al.[20] aprašo skirtingus pozicijų paieškos bei vertinimo funkcijų algoritmus, taip pat pateikia daug jų pritaikymo pavyzdžių.

Pagrindinė ligando įvedimo į baltymą technikos problema yra ta, kad reikia ištirti labai didelį skaičių galimų susijungimo variantų.[8] Ankstesniais laikais tokius tyrimus smarkiai ribojo kompiuteriniai resursai, tačiau dabar kompiuteriai yra pajėgesni. Šiuo metu yra sukurta nemažai programinės įrangos baltymo ir ligando prijungimui modeliuoti, tarp kurių yra ir nemokamai ir viešai prieinamų, tokių kaip „AutoDock" bei „LeDock".[24] Verta paminėti, kad projektavimui pagreitinti yra naudojami supaprastintos struktūros arba molekuliniai vektoriai (angl. *molecular fingerprints*).[2]

1.2.1. Ligando pozicijos komplekse paieškos algoritmai

Kaip jau buvo minėta, reikia ištirti labai didelį ligando pozicijų kiekį. Analizės greitinimui buvo sugalvoti skirtingi kompiuteriniai algoritmai. Juos galima klasifikuoti pagal ignoruojamų parametrų kiekį. Pats paprasčiausias algoritmas atkreipia dėmesį tik į baltymo bei ligando nelanksčias trimates struktūras. Toks algoritmas ieško ligando pozicijas, kurios atitinka „spynos ir rakto" teorijai.[5] Sudėtingesnis algoritmas, vadinantis inkrementiniu konstravimu (angl. *incremental construction*), fragmentuoja ligandą palei jo galinčius suktis ryšius į segmentus.[8] Dar vienas algoritmas vadinamas genetiniu: pasirenkama ligando pozicijų grupė ir įvertinamos jų surišimo su baltymu konstantos, toliau geriausios pozicijos atsitiktinai nežymiai pakeičiamos ir/ar fragmentai pozicijų sumaišomos tarpusavyje, ir taip atsiranda sekanti pozicijų „populiacija". Procesas kartojasi daug kartų. Kituose straipsniuose[8][24] pozicijos paieškos algoritmus dalinama į tris grupes: formų priderinimo (angl. *shape matching*), sistematinės paieškos (angl. *systematic search*) ir stochastinės paieškos (angl. *stochastic search*) algoritmai.

1.2.2. Komplekso surišimo vertinimo funkcijos

Vertinimo funkcija yra antras ligando įvedimo į baltymą etapas: naudojant vertinimo funkciją, galima prognozuoti baltymo ir turimos pozicijos ligando surišimą (arba disociacijos konstantą, K_d). Gera vertinimo funkcija turi būti didesnio nei 1 pK_d tikslumo, greitai ir lengvai skaičiuojama. Berry et al.[3] vadina vertinimo funkcijas silpnąja baltymo–ligando jungimosi vieta, nes tai yra svarbiausia ligando įvedimo į baltymą dalis, kurioje iki šiol yra aptinkama nemažai problemų.

Klasikinės vertinimo funkcijos skirstomos į tris grupes: funkcijos pagrįstos fizine sąveika, empirinės funkcijos ir funkcijos pagrįstos teorinėmis žiniomis.[12] Neseniai atsirado dar vienas vertinimo funkcijų tipas, kurį dažniausiai vadina kaip mašininio mokymosi vertinimo funkcijos. Jos parodo geresnius rezultatus baltymo–ligando jungimosi prognozavimee.[25] Liu et al.[12] rekomenduoja šį tipą vadinti deskriptorių funkcijomis (angl. *descriptor-based*), argumentuojant tuo faktu, kad terminas „mašininis mokymasis" aprašo ne teorinę bazę, bet technologiją. Kitas jų argumentas nurodo, kad teorinėmis žiniomis pagrįstos funkcijos taip pat gali naudoti mašininį mokymąsi.

Vertinimo funkcijos rezultatų atitikimas tikrovei gali būti patikrintas vertinant pačią vertinimo funkciją. Tai daroma tikrinant funkcijos baltymo–ligando kompleksų surišimo prognozes su tikromis tų kompleksų vertėmis, paduodant funkcijai jau išanalizuotų kompleksų rinkinį. Huang et al.[7] aprašo skirtingas vertinimo funkcijos įverčių formules. Viena iš klasikinių formulių yra šaknies vidurkio kvadrato nuokrypis (angl. *root mean square deviation*). Jis skaičiuojamas tarp geriausios prognozuotos ligando pozicijos ir eksperimentiškai nustatytos pozicijos ir užrašomas taip:

$$RMSD(v, w) = \sqrt{\frac{1}{n}\sum_{i=1}^{n} ||v_i - w_i||^2} \tag{3}$$

kur v_i ir w_i yra du koordinačių rinkiniai, o n yra atomų skaičius. Jeigu RMSD reikšmė yra mažesnė arba lygi 2Å, prognozė laikoma sėkminga.[3] Didžiausias RMSD privalumas yra paprastumas, tačiau pagrindinė jo problema yra tame, kad maži ar simetriški ligandai turės mažą RMSD įvertį net nebūdami tinkamoje pozicijoje. Todėl buvo pasiūlyti keletas panašių įverčių kaip reliatyvaus poslinkio paklaida (angl. *relative displacement error*), sąveika paremta tikslumo klasifikacija (angl. *interaction-based acuracy classification*) ir kiti.[7] Visi jie įvertina prognozuotos ligando pozicijos tinkamumą lyginant su duota pozicija.

Kita svarbi vertinimo funkcijos savybė yra nustatyti jungties stiprumą - tai yra atliekama regresijos būdu. Tam naudojama Pearson koreliacija tarp apskaičiuotų balų ir eksperimen-

tinių duomenų.[1] Ji turi sekančią formulę:

$$R_p = \frac{\sum_{k=1}^{N}(x_k - \langle x\rangle)(y_k - \langle y\rangle)}{\sqrt{[\sum_{k=1}^{N}(x_k - \langle x\rangle)^2][\sum_{k=1}^{N}(y_k - \langle y\rangle)^2]}} \tag{4}$$

kur N yra kompleksų skaičius, x_k – eksperimentiškai nustatyta jungties energija, y_k – jungties energija apskaičiuota k-ajam kompleksui.[7]

Toks R įvertis yra aritmetinis visų kompleksų vidurkis. Šiam įverčiui pagerinti buvo pasiūlytas Spearman koreliaijos koeficientas, kurį apibūdina ši formulė:

$$R_s = 1 - \frac{6\sum_{k=1}^{N} d_k^2}{N(N^2-1)} \tag{5}$$

Čia d_k yra dviejų vertinimų skirtumas k-ajam kompleksui.

Trečioji svarbi vertinimo funkcijos savybė yra gebėjimas atrinkti tinkamus ligandus iš didelės jų gausos, kas dar vadinama virtualia atranka. Vienas iš šio proceso įverčių yra praturtinimo testas (angl. *enrichment test*). Jis yra populiarus dėl savo paprastumo,[1][25] o jo formulė atrodo taip:

$$EF_{x\%} = \frac{\frac{\text{aktyvieji ligandai}(x\%)}{\text{visi ligandai}(x\%)}}{\frac{\text{aktyvieji ligandai}(100\%)}{\text{visi ligandai}(100\%)}} \tag{6}$$

Kuo didesnis praturtinimo įvertis, tuo geresnė vertinimo funkcija.

Kitas įvertis, taikomas klasifikacijos metodams, yra plotas po kreive (angl. *area under curve* arba sutrumpintai AUC), kur kreivė yra ROC (angl. *receiver operating characteristic*) kreivė. Jis parodo, ant kiek modelis geba atskirti aktyvius ir neaktyvius ligandus, kur AUC = 0,5 reiškia visiškai atsitiktinus spėjimus, o AUC = 1 reiškia idealų modelį. Šis metodas tinka, jei aktyvių ir neaktyvių ligandų skaičius rinkinyje yra maždaug panašus.[7]

Vertinimo funkcijos patikrinimas dažnai vyksta naudojant kompleksus–„apgavikus" (angl. *decoys*). Jie yra panašūs į realus aktyviai surištus kompleksus, tačiau iš tikrųjų nėra (arba bent jau neturi būti) aktyvūs. Populiariausios „apgavikų" duomenų bazės yra „DUD" (angl. *A Directory of Useful Decoys*) bei „DUD-E" (angl. *A Database of Useful Decoys: Enhanced*).[3] Tokie kompleksai yra generuojami kompiuteriniais metodais, dėl ko gali atsirasti šališkumas (angl. *bias*), kaip teigia Reau et al.[17] Šis šališkumas gali neigiamai paveikti ypač neuroniniu tinklu pagrįstos vertinimo funkcijos veikimą, nes tokios funkcijos savarankiškai atranda kompleksų klasifikacijos parametrus. Tokio šališkumo pavyzdys yra pateiktas straipsnyje[4], kurio autoriai teigia, kad „DUD-E" duombazėje yra analoginis šališkumas (angl. *analogue bias*), dėl ko neuroninio tinklo vertinimo funkcijos gauna ypač

aukštus įverčius (AUC > 0,9). Autoriai paruošė konvoliucinių neuroninių tinklų modelį ir apmokė jį naudojant du skirtingus rinkinius: pilnus kompleksus ir tik vienus ligangus iš „DUD-E". Rezultatai buvo labai panašūs ir abu gavo aukštą AUC įvertį, kas parodė, jog modelis klasifikuoja kompleksus neskiriant jokio dėmesio į baltymo receptoriaus struktūrą. Todėl autoriai siūlo stengtis nenaudoti kompiuteriu sugeneruotus „apgavikus", o geriau naudoti eksperimentiškai analizuotus neaktyvius kompleksus.

1.3. Mašininio mokymosi taikymas

Mašininio mokymosi algoritmai pasižymi tuo, kad įgauna savo funkcionalumą mokymosi metu ir pati atranda dėsningumus, kurių žmogui būtų sunku ir per ilgai nustatyti.[21] Iš kitos pusės, mašininio mokymosi funkcijos pasirinktų parametrų esmė ne visada yra žmogui suprantama. Šis efektas dar vadinasi „juodos dėžės" efektu (angl. *„black box"*) ir dažniausiai pasireiškia taikant dirbtinius neuroninius tinklus,[12] apie kuriuos bus kalbama 1.3.1 skyriuje. Mašininis mokymasis gali būti prižiūrėtas arba ne (angl. *supervised* ir *unsupervised*). Pirmu atveju funkcijai yra nurodoma, ar mokymosi metu gautas rezultatas atitinka tikrovei, ir naudojant šią informaciją, funkcija keičia savo parametrus kad padidintų tikslumą. Neprižiūrėto mokymosi atveju, funkcija bando pati klasifikuoti turimus duomenis į grupes pagal parametrus, kuriuos pati pasirenka.[19] Baltymo–ligando komplekso surišimo vertinimo atveju, mašininis mokymasis turi būti prižiūrėtas.

Pagrindinis mašininio mokymosi veikimo bendri etapai yra:

- Didelės tikrų duomenų duombazės pasirinkimas: kuo daugiau duomenų ir kuo jie kokybiškesni bei apima daugiau galimų variantų, tuo geriau galima apmokinti funkciją.[21]
- Duomenų dalijimas į apmokymo ir testavimo rinkinius (dažnai 70/30 ar 80/20 santykiu). Apmokymo rinkinys būna didesnis dėl to, kad apmokymui reikia kuo daugiau duomenų.
- Funkcijos apmokymas naudojant tik apmokymo rinkinį: algoritmas keičia savo parametrus taip, kad kuo tiksliau atspėtų apmokymo duomenis, randa bendrus dėsningumus;
- Funkcijos patikrinimas ant testavimo duomenų: jei atsiranda didelis procentas blogai prognozuotų reikšmių, reiškia funkcija yra permokyta (angl. *overfitted*). Jei teisingai prognozuotų reikšmių procentas panašus į gautą po funkcijos apmokymo ir kartu pakankamai aukštas, funkcija yra paruošta naudojimui ar tolimesniam testavimui.

Mašininio mokymosi algoritmas gali spręsti klasifikacijos arba regresijos problemą.[10] Klasifikacijos problemos pavyzdys būtų diagnozės išvedimas žinant tam tikrus parametrus - pavyzdžiui pacientų skyrimas į sergančius ir nesergančius pagal medicininės analizės rezultatus. Tuo metu regresijos problema yra susieta su tam tikro skaičiaus keitimu, o tokios problemos pavyzdys yra butų kainų prognozavimas. Prognozuojant komplekso K_d, galima taikyti abu sprendimus: klasifikacijos atveju tai būtų tiesiog ligandų sužymėjimas, parodantis ar jie susiriš su baltymų; regresijos atveju būtų bandoma prognozuoti tikslią K_d vertę. Atitinkamai, klasifikuojančiai funkcijai užteks duombazės, kur yra pateikti galintys ir negalintys susirišti kompleksai (tam dažnai naudojami kompleksai–„apgavikai"). Tuo metu regresijos funkcijai bus reikalinga kompleksų duombazė su nustatytomis K_d reikšmėmis.

Mašininio mokymosi algoritmai gali persimokyti.[16] Tai reiškia, kad jie gerai prognozuos duomenis, su kuriais mokėsi, bet blogai prognozuos kitus duomenis iš tos pačios srities. Taip atsitinka, kai modeliui pateikti duomenys neapima visų galimų variantų. Pavyzdžiui, modelis gali prisitaikyti prie tam tikros baltymų šeimos ir blogai prognozuoti kitos baltymų šeimos sąryšį su ligandu. Kitaip sakant, modelis neišmoko tam tikrų taisyklių, bet tiesiog „iškalė" duomenis mintinai. Permokymą galima pastebėti, kai mokymosi rinkinio prognozavimas pavyksta kuo puikiausiai, tuo pačiu metu su nepriklausomais duomenimis prognozavimas tampa labai klaidingas. Modelis tampa specifinis.[10] Tai galima taisyti keliais būdais, vienas iš kurių yra nežymiai pakeistų duomenų pridėjimas (angl. *data augmentation*).[16] Pakeitimai turi būti nereikšmingi, kad modelis nepradėtų mokytis ant klaidingų duomenų. Pavyzdžiui jei kalbama apie konvoliucinius neuroninius tinklus, galima tą patį paveiksliuką pasukti arba pridėti nežymaus triukšmo. Baltymų-ligandų kompleksų atveju, jei modelis mokosi iš kompleksų trimačių struktūrų, galima tas struktūras įvairiai pasukti.

Kitas būdas kovoti su permokymu yra kryžminės kontrolės naudojimas (angl. *cross-validation*).[10][25] Šiuo atveju duomenys dalinami į k grupes. Toliau modelis mokosi naudojant $k-1$ grupių duomenis, o paskutinė grupė veikia kaip testavimo rinkinys. Šis procesas kartojasi k kart, kiekvieną kartą išrenkama vis kita grupė testavimui. Taip įvyksta tolygus mokymasis su visais duomenimis.[25]

Nuostolių funkcijos reguliarizacija taip pat yra veiksmingas būdas mažinti permokymą. Pačios žinomiausios yra L1 ir L2 reguliarizacijos. L1 algoritmas (dar žinomas kaip Lasso regresija) tiesiogiai mažina parametrų svorius. Parametrai, kurie turi mažesnį svorį, visai išnyksta. Taip modelis įgyja kontrastą, nes daugiau išryškėja lemtingi parametrai; tai puikiai tinka klasifikacijos metu atrinkti reikšmingus parametrus. L2 algoritmas (dar žinomas kaip gūbrio regresija, angl. *ridge regression*) mažina parametrų kvadratinius svorius. Tuomet visi

svoriai mažėja tolygiai, modelis tampa labiau generalizuotas.[21]

1.3.1. Dirbtiniai neuroniniai tinklai

Dirbtiniai neuroniniai tinklai yra mašininio mokymosi algoritmai, paremti gamtiniais neuroniniais tinklais.[16] Dirbtinis neuroninis tinklas sudarytas iš neuronų sluoksnių, kurių būna du ir daugiau ir jie nuosekliai sujungti. Išoriniai sluoksniai yra „matomi", o kiti sluoksniai yra paslėpti tarp jų ir vadinasi giliaisiais.[19] Ryšiai tarp neuronų turi svorius, kurie parodo to ryšio savotišką svarumą. Neuronai gali atlikti skirtingus pakeitimus ir būna įvairiai sujungti priklausant nuo tinklo tipo ir uždavinio. Neuroninių tinklų uždaviniai gali būti tam tikro „rašto" identifikavimas, signalo apdorojimas ir kiti, o galinis neuronų sluoksnis generuoja „atsakymą" - dažniausiai tai būna paduotos į tinklą informacijos klasifikavimas.[16]

Dirbtinio neuroninio tinklo ypatumas yra tame, kad žmogus gali logiškai interpretuoti tik pirmą ir paskutinį tinklo sluoksnius. Paprasčiausi neuroniniai tinklai turi tik du sluoksnius, tačiau dabar populiarėja gilieji neuroniniai tinklai (angl. *Deep Neural Networks*), kurie gali turėti daugiau nei tris sluoksnius. Kaip būtent daugiasluoksnio neuroninio tinklo modelis atpažįsta tam tikrus požymius, iki šiol nėra tiksliai nustatyta. Iš to yra iškeliama svarbi problema: nesuprantama iki galo, kaip būtent veikia dirbtiniai neuroniniai tinklai. Šis efektas, kaip buvo minėta anksčiau, vadinamas „juodosios dėžės" efektu. Nepaisant to, gilieji neuroniniai tinklai plačiai naudojami bioinformatikoje.[4][21]

Šiuo momentu egzistuoja daug skirtingų neuroninio tinklo išsidėstymų, pavyzdžiui rekursiniai (angl. *recurrent neural network*) arba konvoliuciniai (angl. *convolutional network*). Rekursinis tinklas turi neuronus, kurie savo išeinantį signalą vėl gauną praėjus fiksuotam laiko tarpui. Tai yra naudinga tyrinėjant sistemas, kur svarbu išlaikyti kontekstą (pavyzdžiui darbui su tekstu, kur prieš tai buvę žodžiai ir sakiniai yra reikšmingi sekančio žodžio prasmei). Šis principas buvo dar patobulintas ir 1997 m. išrastas ilgos trumpos atminties algoritmas (angl. *long short-term memory*), kurio elementai gali nuspręsti, ar verta laikyti tam tikrą reikšmę neurono atmintyje. Rekursiniai tinklai ir natūralios kalbos apdorojimas (angl. *natural language processing*) buvo analizuoti Jastrzębski et al.[9] kaip būdas spręsti bioinformatikos klasifikacijos problemas, kai duomenys apie molekules pateikiami SMILES formatu.

Konvoliuciniai neuroniniai tinklai dažniausiai naudojami paveikslėlių atpažinimui. Jie gali „atpažįsti" tam tikrus požymius (linijas, figūras, spalvas ir t.t.) ir pagal jas klasifikuoti paveikslėlį. Taip pat konvoliuciniai neuroniniai tinklai gerai tinka analizuoti daugiamates

struktūras, pavyzdžiui baltymo molekulės struktūrą, ir yra perspektyvūs ligando įvedimo į baltymo vertinimui.[6] Ragoza et al.[16] sėkmingai panaudojo konvoliucinį neuroninį tinklą baltymo–ligando kompleksų surišimo vertinimui. Jie aprašo konvoliucinių tinklų veikimo principą: kiekvienas tinklo sluoksnis bando atpažinti paveikslėlyje tam tikrą figūrą, ir kuo daugiau tinklas turi sluoksnių, tuo sudėtingesnius „raštus" jis gali identifikuoti ir klasifikuoti. Daugiau apie konvoliucinių neuroninių tinklų taikymą vertinimo funkcijose yra kalbama 1.3.3 skyriuje.

Šiame darbe bus naudojami konvoliuciniai neuroniniai tinklai vertinimo funkcijos kūrimui.

1.3.2. Ligandų virtuali atranka

Ligando įvedimas į baltymą daugiausia yra naudojamas vaistų projektavime. Prijungimų palyginimas leidžia nustatyti, kaip sąveikaus įvairios molekulės, ir taip atrenkami ligandai, kurie tikimiausiai bus veiksmingi vaistai. Savaime suprantama, kad vienos kompiuterinės analizės neužtenka: kiekviena atrinkta molekulė bus tiriama *in vitro*, taip pat kiekvienas vaistas turi praeiti klinikinius tyrimus. Programinės įrangos panaudojimas molekulių prijungime yra esminis, kai norima sumažinti tiriamų molekulių skaičių iš pradinės ligandų duombazės.[15] Šis procesas vadinamas virtualia atranka (angl. *virtual screening*).

Virtuali atranka gali vykti dviems būdais: tai gali būti ligandų atranka pagal jų fiziko–chemines savybes lyginant su jau žinomais aktyviais ligandais; arba ligandus galima atrinkti pagal jų trimates struktūras, kai receptoriaus struktūra taip pat yra nustatyta. Struktūra paremta virtuali atranka dažniau yra tikslesnė negu paremta tiesiog fiziko–cheminėmis ligandų savybėmis.[15]

1.3.3. Mašininio mokymosi vertinimo funkcijos ir jų pavyzdžiai

Kaip jau buvo minėta anksčiau, pagrindinis skirtumas tarp klasikinių ir mašininio mokymosi vertinimo sistemų yra jų galutinio algoritmo įgavimo būdas. Klasikinio metodo vertinimo sistema turi turėti teorinę bazę – žmogaus išvestą formulę. Mašininio mokymosi vertinimo sistema įgauna savo funkcionalumą mokymosi metu, pati atranda dėsningumus ir supranta, į kuriuos molekulės duomenis reikia atkreipti dėmesį. Mašininio mokymosi metodas yra pagrįstas didelio duomenų kiekio nagrinėjimu ir vertinimo funkcijos išvedimu keičiant funkcijos parametrus iki tol, kol funkcija gali kuo tiksliau prognozuoti baltymo–ligando komplekso jungimąsi.

Viena iš populiariausių ligando įvedimo į baltymą programų yra „Autodock Vina"[22], pristatyta 2010 metais. Ji yra nemokama ir viešai prieinama. Jos vertinimo funkcija yra dažniausiai priskiriama prie empirinių funkcijų tipo, tačiau jos autoriai teigia, kad ji yra daugiau pagrįsta mašininiu mokymusi. Šiandien „Autodock Vina" galima aptikti tokiose plačiai naudojamose programose, kaip „UCSF Chimera", bei daugelyje mokslinių straipsnių.

Moderniausi vertinimo funkcijų modeliai yra kuriami naudojant dirbtinius neuroninius tinklus, tačiau puikiai veikia ir kiti mašininio mokymosi algoritmai, tokie kaip atsitiktinių miškų (angl. *Random Forest*) ar atraminių vektorių mašinų (angl. *Support–Vector Machines*) algoritmai. Pavyzdžiui, Wong et al. [26] panaudojo atraminių vektorių mašinų algoritmą ligando įvedimui į baltymą. Šis algoritmas yra skirtas rasti $N-1$ dimensijų (kai yra N savybių) hiperplaną - tai yra atskirti duomenų taškus į klasterius hiperplano pagalba. Wong savo metodui panaudojo 29 baltymo–ligando komplekso savybes, tokios kaip jungties energija, sekos konservatyvumas, atstumas tarp molekulių ir t.t. Taip buvo nustatyta, kurie iš išrinktų baltymų ertmių arba „kišenių" turi didžiausią tikimybę pririšti ligandus. Tais pačiais metais buvo sukurta panašaus veikimo vertinimo funkcija „ID-score", kuri pagrįsta atraminių vektorių regresija ir kiekvienam kompleksui išskiria net 50 parametrų.[11] Atsitiktinių miškų algoritmas buvo naudojamas „RF-score" vertinimo funkcijoje, po kurios buvo sumodeliuotos „RF-score–2", „RF-score–3" ir „RF-score–VS" funkcijos. Paskutinė jų yra iki šiol stiprus konkurentas neuroninio tinklo funkcijoms.[25]

Pagal Ain et al.,[1] pirma dirbtinio neuroninio tinklo baltymo–ligando surišimo vertinimo funkcija buvo pristatyta 2008 metais. Konvoliuciniais neuroniniais tinklais paremtos vertinimo funkcijų pavyzdžiai yra „AtomNet" (2013)[23], „DeepVS" (2016)[15], „Gnina" (2017)[16] ir „Pafnucy" (2018)[21]. Šiame darbe yra akcentuojamas „DeepVS" modelis, ir jo algoritmas yra naudojamas šio darbo vertinimo funkcijos kūrimui. Jis buvo parinktas dėl jo aukšto įvertinimo palyginus su senesnėmis vertinimo funkcijomis bei aiškaus ir nuoseklaus aprašo. Taip pat ši vertinimo funkcija yra viešai prieinama adresu https://github.com/JanainaCruz/DeepVS.

1.3.4. Duomenų paruošimas pateikimui į modelį

Mašininio mokymosi modeliui yra ypač svarbu paruošti didelį ir kokybišką duomenų rinkinį, šiuo atveju baltymo–ligando kompleksų rinkinį su nustatyta K_d (tiksli reikšmė reikalinga norint spręsti regresijos užduotį, o klasifikacijos užduočiai užtenka binarinio indikatoriaus apie komplekso aktyvumą). Tokius rinkinius galima gauti iš skirtingų duombazių; vieni po-

puliariausių yra „RCSB PDB", „CSAR" ir t.t.[16] Taip pat yra mažesnės duombazės, skirtos būtent proteino–ligando kompleksams ir jų vertinimo funkcijoms, tokios kaip „PDBbind". Ligandų paieškai gali būti naudojama „ZINC", komercinių ligandų duomenų bazė.

Pieš modeliui apdorojant duomenis, pačius duomenis būtinai reikia tinkamai paruošti. Šiuo atveju reikia išrinkti molekulių (baltymų ir ligandų) užrašymo būdą. Dažnai tam yra naudojami savybių vektoriai (molekuliniai vektoriai, angl. *fingerprints*).[2] Molekuliniai vektoriai yra skaičiais užrašytų savybių rinkiniai. Priklausomai nuo vektoriaus tipo, jis gali būti įvairaus ilgio ir nurodyti skirtingus parametrus. Yra svarbu, kad kiekvienas kompleksas būtų užrašytas tuo pačiu formatu. Vektoriaus dydis ir informacija įtakoja galutinio modelio apmokymo greitį ir efektyvumą, nes vienos savybės gali turėti įtakos surišimo prognozavimui, kai kitos tik sulėtins skaičiavimus. Ballester et al.[2] teigia, kad didelis cheminių savybių tikslumas nebūtinai generuoja tikslesnį rezultatą. Panašus efektas buvo pastebėtas Ragoza et al.[16]

Kalbant apie ligando bei receptoriaus molekulių trimates struktūras, ligando įvedimui į baltymą pastarajam yra iškarpomas ir naudojamas aktyvusis centras proceso greitinimui. Paskui gautą trimatį molekulės gabaliuką galima išreikšti kaip keturmatį tensorių, kuris yra sudarytas iš taškų (atomų); kiekvienas taškas turi tris koordinates ir savybių vektorių. Jame užrašomas atomo tipas, hibridizacija, ryšių kiekiai, dalinis krūvis ir t.t.[21]

2. METODAI

2.1. Duomenų apie molekules surinkimas

2.2. Lingadų įvedimas į batymą

Naudojama programa - LeDock

Literatūra

[1] AIN, Q.U.; *et al. Machine-learning scoring functions to improve structure-based binding affinity prediction and virtual screening.* Wiley Interdisciplinary Reviews. Computational Molecular Science, 2015. 5, 6, 405–424. ISSN 1759-0876.

[2] BALLESTER, P.J.; SCHREYER, A.; BLUNDELL, T.L. *Does a more precise chemical description of protein-ligand complexes lead to more accurate prediction of binding*

affinity? Journal of Chemical Information and Modeling, 2014. 54, 3, 944–955. ISSN 1549-960X.

[3] BERRY, M.; FIELDING, B.; GAMIELDIEN, J. *Practical Considerations in Virtual Screening and Molecular Docking. Emerging Trends in Computational Biology, Bioinformatics, and Systems Biology*, Elsevier, 487–502. ISBN 978-0-12-802508-6, 2015.

[4] CHEN, L.; *et al. Hidden Bias in the DUD-E Dataset Leads to Misleading Performance of Deep Learning in Structure-Based Virtual Screening* [Interaktyvus]. 2019. Prieiga per: https://chemrxiv.org/articles/Hidden_Bias_in_the_DUD-E_Dataset_Leads_to_Misleading_Performance_of_Deep_Learning_in_Structure-Based_Virtual_Screening/7886165.

[5] DU, X.; *et al. Insights into Protein-Ligand Interactions: Mechanisms, Models, and Methods.* International Journal of Molecular Sciences, 2016. 17, 2. ISSN 1422-0067.

[6] HOCHULI, J.; *et al. Visualizing convolutional neural network protein-ligand scoring.* Journal of Molecular Graphics and Modelling, 2018. 84, 96–108. ISSN 10933263.

[7] HUANG, S.Y.; GRINTER, S.Z.; ZOU, X. *Scoring functions and their evaluation methods for protein–ligand docking: recent advances and future directions.* Physical Chemistry Chemical Physics, 2010. 12, 40, 12899. ISSN 1463-9076, 1463-9084.

[8] HUANG, S.Y.; ZOU, X. *Advances and challenges in protein-ligand docking.* International Journal of Molecular Sciences, 2010. 11, 8, 3016–3034. ISSN 1422-0067.

[9] JASTRZĘBSKI, S.; LÉSNIAK, D.; CZARNECKI, W.M. *Learning to SMILE(S)* [Interaktyvus]. [Interaktyvus], 2018. Prieiga per: http://arxiv.org/abs/1602.06289.

[10] LEHR, D.; OHM, P. *Playing with the Data: What Legal Scholars Should Learn about Machine Learning* [Interaktyvus]. U.C. Davis Law Review, 2017. 51, 653. Prieiga per: https://heinonline.org/HOL/Page?handle=hein.journals/davlr51&id=667&div=&collection=.

[11] LI, G.B.; *et al. ID-Score: A New Empirical Scoring Function Based on a Comprehensive Set of Descriptors Related to Protein–Ligand Interactions.* Journal of Chemical Information and Modeling, 2013. 53, 3, 592–600. ISSN 1549-9596.

[12] LIU, J.; WANG, R. *Classification of current scoring functions.* Journal of Chemical Information and Modeling, 2015. 55, 3, 475–482. ISSN 1549-960X.

[13] MATULIS, D. *Carbonic Anhydrase As Drug Target: Thermodynamics and Structure of Inhibitor Binding*. Springer, 2019. ISBN 978-3-030-12780-0.

[14] MENG, X.Y.; *et al. Molecular Docking: A powerful approach for structure-based drug discovery*. Current computer-aided drug design, 2011. 7, 2, 146–157. ISSN 1573-4099.

[15] PEREIRA, J.C.; CAFFARENA, E.R.; DOS SANTOS, C.N. *Boosting Docking-Based Virtual Screening with Deep Learning*. Journal of Chemical Information and Modeling, 2016. 56, 12, 2495–2506. ISSN 1549-960X.

[16] RAGOZA, M.; *et al. Protein–Ligand Scoring with Convolutional Neural Networks*. Journal of Chemical Information and Modeling, 2017. 57, 4, 942–957. ISSN 1549-9596, 1549-960X.

[17] RÉAU, M.; *et al. Decoys Selection in Benchmarking Datasets: Overview and Perspectives*. Frontiers in Pharmacology, 2018. 9. ISSN 1663-9812.

[18] SCHIEBEL, J.; *et al. Intriguing role of water in protein-ligand binding studied by neutron crystallography on trypsin complexes*. Nature Communications, 2018. 9. ISSN 2041-1723.

[19] SCHMIDHUBER, J. *Deep learning in neural networks: An overview*. Neural Networks, 2015. 61, 85–117. ISSN 08936080.

[20] SETHI, A.; *et al. Molecular Docking in Modern Drug Discovery: Principles and Recent Applications* [Interaktyvus]. Drug Discovery and Development - New Advances, 2019. Prieiga per: https://www.intechopen.com/online-first/molecular-docking-in-modern-drug-discovery-principles-and-recent-applications.

[21] STEPNIEWSKA-DZIUBINSKA, M.M.; ZIELENKIEWICZ, P.; SIEDLECKI, P. *Development and evaluation of a deep learning model for protein–ligand binding affinity prediction*. Bioinformatics, 2018. 34, 21, 3666–3674. ISSN 1367-4803.

[22] TROTT, O.; OLSON, A.J. *AutoDock Vina: improving the speed and accuracy of docking with a new scoring function, efficient optimization and multithreading*. Journal of computational chemistry, 2010. 31, 2, 455–461. ISSN 0192-8651.

[23] WALLACH, I.; DZAMBA, M.; HEIFETS, A. *AtomNet: A Deep Convolutional Neural Network for Bioactivity Prediction in Structure-based Drug Discovery* [Interaktyvus]. 2015. ArXiv: 1510.02855.

[24] WANG, Z.; *et al. Comprehensive evaluation of ten docking programs on a diverse set of protein–ligand complexes: the prediction accuracy of sampling power and scoring power.* Physical Chemistry Chemical Physics, 2016. 18, 18, 12964–12975. ISSN 1463-9076, 1463-9084.

[25] WÓJCIKOWSKI, M.; BALLESTER, P.J.; SIEDLECKI, P. *Performance of machine-learning scoring functions in structure-based virtual screening.* Scientific Reports, 2017. 7. ISSN 2045-2322.

[26] WONG, G.Y.; LEUNG, F.H.F.; LING, S.H. *Predicting protein-ligand binding site using support vector machine with protein properties.* IEEE/ACM transactions on computational biology and bioinformatics, 2013. 10, 6, 1517–1529. ISSN 1557-9964.