4. STROJNO UČENJE

Avtor: Dejan Mirčetić

Veliko je vprašanj o tem, kaj je strojno učenje (ML – Machine Learning). Ali gre za dejanski proces, v katerem se stroji sami učijo iz zunanjega okolja, ali za formaliziran proces z matematičnimi algoritmi, ki računalnikom omogočajo, da "ugotovijo" pravila v zunanjem svetu? Katera orodja uporablja ML? Kako je videti tipičen pretok podatkov v cevovodu ML? Ali se uporablja v tradicionalnih panogah, ne le v panogah, povezanih z informacijsko tehnologijo in internetom? Kje je mesto ML v okviru poslovanja? Kako ga sistematično uporabiti za reševanje poslovnih vprašanj? Ali obstaja kakšna arhitektura, kako jo uporabiti v nadzornih organih?

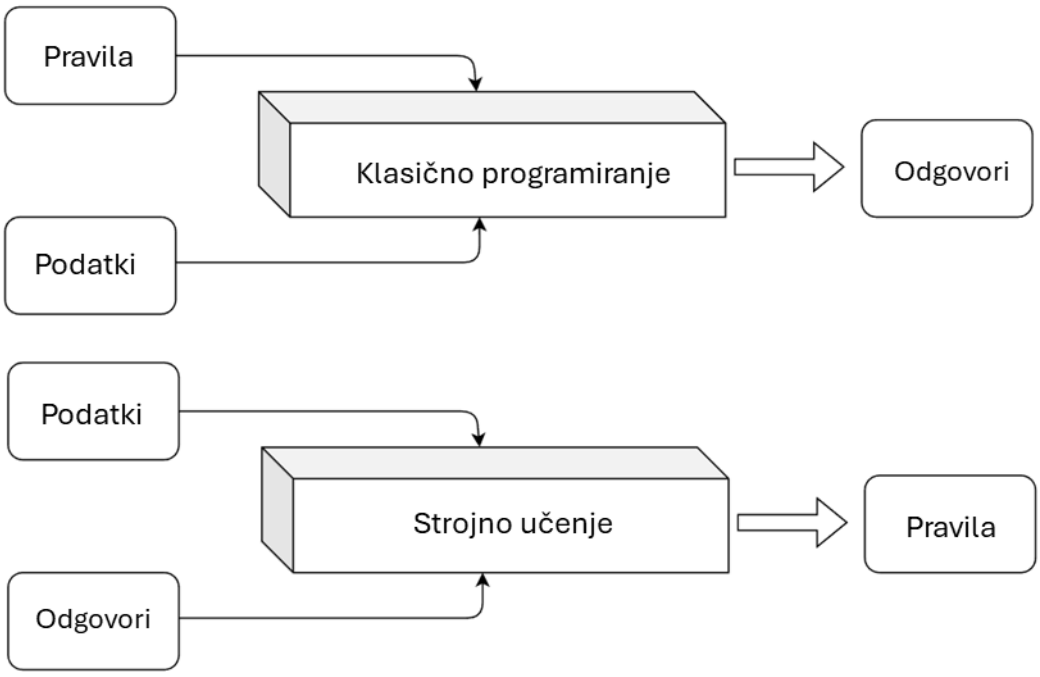
Na ta in podobna vprašanja bomo poskušali odgovoriti v naslednjem poglavju, ki ga bomo zaključili s primerom študije primera uporabe algoritmov ML v verigi preskrbe s hrano.

4.1. Kaj je strojno učenje?

Strojno učenje je disciplina, ki se osredotoča na dve medsebojno povezani vprašanji: Kako lahko zgradimo računalniške sisteme, ki se samodejno izboljšujejo z izkušnjami? in Kateri so temeljni statistični računalniško-informacijsko teoretični zakoni, ki veljajo za vse učne sisteme, vključno z računalniki, ljudmi in organizacijami? Študij strojnega učenja je pomemben tako zaradi obravnavanja teh temeljnih znanstvenih in inženirskih vprašanj kot tudi zaradi zelo praktične računalniške programske opreme, ki je bila ustvarjena in uporabljena v številnih aplikacijah (Jordan in Mitchell, 2015).

ML izhaja iz tega vprašanja: ali lahko računalnik preseže "tisto, kar mu znamo naročiti", in se sam nauči, kako opraviti določeno nalogo? Ali nas lahko računalnik preseneti? Namesto da bi programerji ročno oblikovali pravila za obdelavo podatkov, bi se lahko računalnik samodejno naučil teh pravil s pregledovanjem podatkov? To vprašanje odpira vrata novi paradigmi programiranja (Chollet, 2021).

ML omogoča temeljno spremembo paradigme programiranja (slika 4.1). Pri klasičnem programiranju programer vnaša pravila (program), podatki pa se analizirajo in obdelujejo v skladu s temi pravili. Rezultat je, da so na koncu na voljo odgovori. Po drugi strani pa pri ML človeški programer vnaša podatke s pričakovanimi odgovori na podlagi podatkov in rezultatom pravil.



Slika 41Klasično programiranje v primerjavi s sistemom strojnega učenja

Vir: Chollet (2021).

Klasično programiranje bi lahko razumeli kot imperativno programiranje, saj programer vnaprej določi vsa pravila in se koda izvaja v skladu z njimi, medtem ko bi lahko ML razumeli kot deklarativno programiranje, kjer izrazimo cilje na višji ravni ali opišemo pomembne omejitve in se zanašamo na matematične algoritme, ki se odločijo, kako in/ali kdaj to prenesti v dejanja.

Danes je ML temelj neštetih pomembnih aplikacij, vključno s spletnim iskanjem, preprečevanjem neželene elektronske pošte, prepoznavanjem govora, priporočanjem izdelkov in drugimi (Ng, 2017). Številni razvijalci sistemov umetne inteligence se zdaj zavedajo, da je za številne aplikacije veliko lažje usposobiti sistem tako, da mu pokažemo primere želenega vhodno-izhodnega vedenjakot pa ga ročno programirati s predvidevanjem želenega odziva za vse možne vhode. Učinek ML se je pokazal tudi v računalništvu in v številnih panogah, ki ukvarjajo z vprašanji, ki zahtevajo veliko podatkov, kot so storitve za potrošnike, diagnosticiranje napak v kompleksnih sistemih in nadzor logističnih verig (Jordan in Mitchell, 2015).

4.2. Osnove in teoretične predpostavke ML

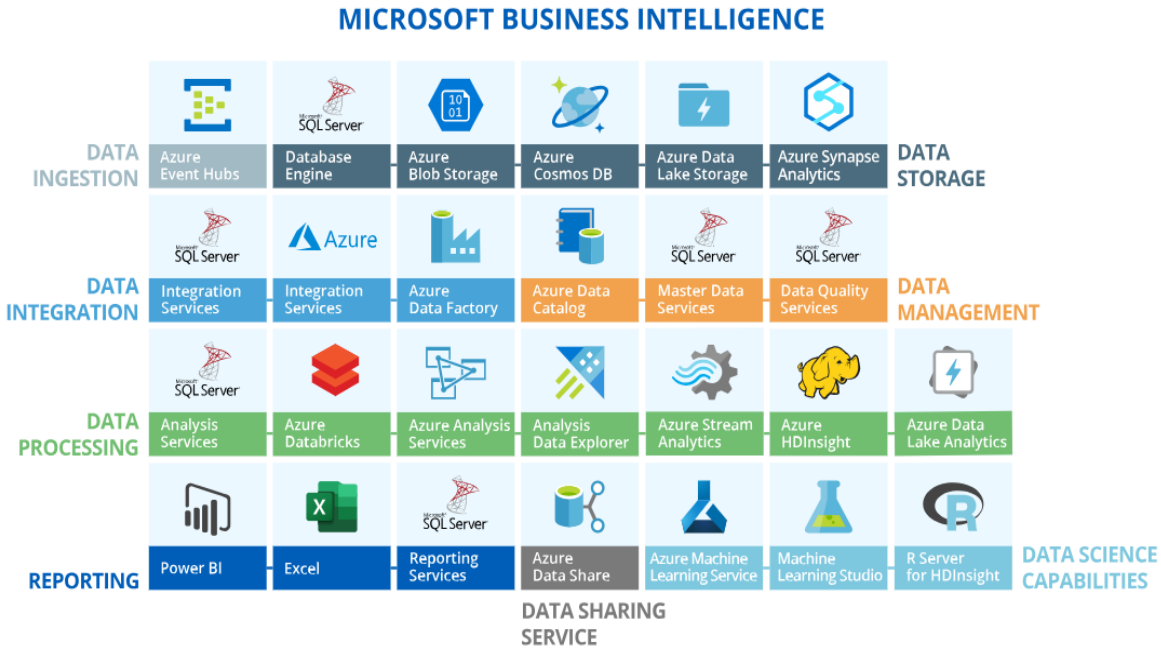
Ozadje ML je v matematiki, natančneje v statistiki. Zato ML uporablja teoretično ozadje in algoritme, razvite v **statističnem učenju**, pri čemer se razpravlja tudi o tem, ali je ML samostojno področje ali pa je le del statistike. V praksi algoritmi ML običajno nimajo določene stopnje matematične togosti in včasih zlahka presežejo nekatere matematične omejitve, ki so prisotne v statistiki. Algoritmi ML na primer pri optimizaciji koeficientov v parametričnih algoritmih ne posvečajo veliko pozornosti intervalom zaupanja, čeprav je to ena najpomembnejših tem v statistiki. Na splošno **se ML in statistika zelo prekrivata**, nekateri najvidnejši ustvarjalci algoritmov ML in profesorji pa trdijo, da je ML le del statistike (Hastie et al., 2009). Kljub temu je ML, ki je samostojno področje ali del statistike, sestavljen iz več korakov pri pridobivanju znanja iz podatkov. Splošnega soglasja o teh korakih ni, na splošno pa jih je mogoče predstaviti kot pretvorbo različnih virov podatkov v spoznanja poslovne inteligence.

V poslovnem kontekstu so modeli ML neuporabni brez ustrezne podpore pri predobdelavi podatkov, podatkovnem rudarjenju in uporabi spoznanj v dejanskih procesih. Zato ustvarjanje algoritmov ML brez možnosti posodabljanja modela in uporabe njegovih rezultatov za dejanski proces odločanja sodobnim podjetjem ne prinaša nobene vrednosti. V skladu s tem je v sodobni poslovni analitiki kvantitativni postopek ML običajno del delovnega procesa poslovne inteligence. Natančneje, je del pomembnih podprocesov poslovne inteligence (podatkovna znanost in podatkovna analitika del poslovne inteligence). Podrobnosti o vlogi ML v teh procesih in samih procesih ustvarjanja vrednosti za poslovanje s pomočjo ML bodo navedene v naslednjem podpoglavju.

4.3. Poslovna inteligenca in ML v SC

Poslovno obveščanje v kontekstu SC je postopek oblikovanja zaključkov o opazovanih procesih SC na podlagi modeliranja podatkov iz teh procesov. Večinoma temelji na statistiki, vendar so v igri tudi druga matematična področja: operacijske raziskave, linearna algebra, fuzzy logika (v primeru, ko je podatkov malo ali jih ni), numerična optimizacija, metahevristika itd. Poleg tega postajajo pomemben vidik za analizo podatkov in pripravo zaključkov tudi nove prelomne tehnologije: **strojno učenje, umetna inteligenca, digitalni dvojčki, smartizacija, živi laboratoriji** itd.

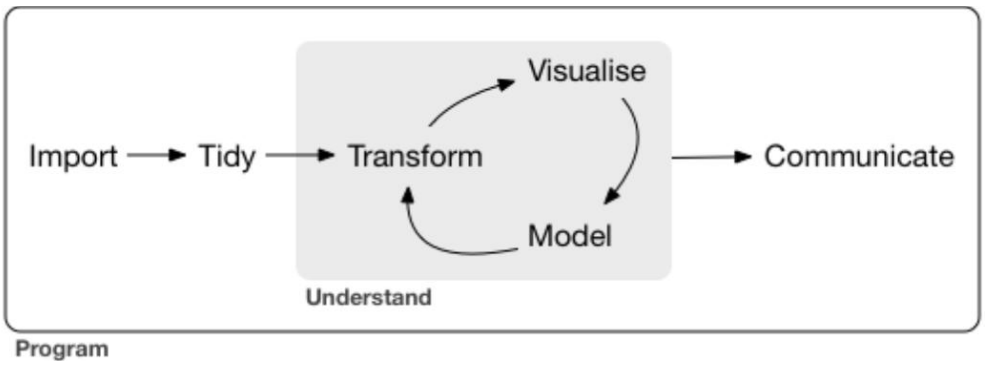
Strogo organiziranih postopkov, kako naj bi bili organizirani poslovna inteligenca postopek in delovni tokovi ML, ni, vendar v praksi in literaturi obstaja nekaj uporabnih smernic, ki so se izkazale za uspešne pri izvajanju analize. Postopek izvajanja poslovne inteligence se razlikuje tudi glede na izvor programske opreme, ki se uporablja za analizo. Microsoft na primer prek svojega kanala Microsoft Business Intelligence package ponuja več orodij, ki izvajajo različne naloge: **vnos podatkov, shranjevanje podatkov, integracija podatkov, upravljanje podatkov, obdelava podatkov, poročanje, izmenjava podatkov in podatkovna znanost** (slika 4.2).



Slika 4.2 Arhitektura Microsoftove poslovne inteligence

Vir: ScienceSoft (n.d.).

V dani arhitekturi se postopki ML uporabljajo samo na ravni podatkovne znanosti z več orodji: Azure ML services, ML studio in R Server for HDInsight. Splošni postopek, kako se analiza podatkov v okviru ML izvaja v strežniku R, je predstavljen na sliki 4.3.

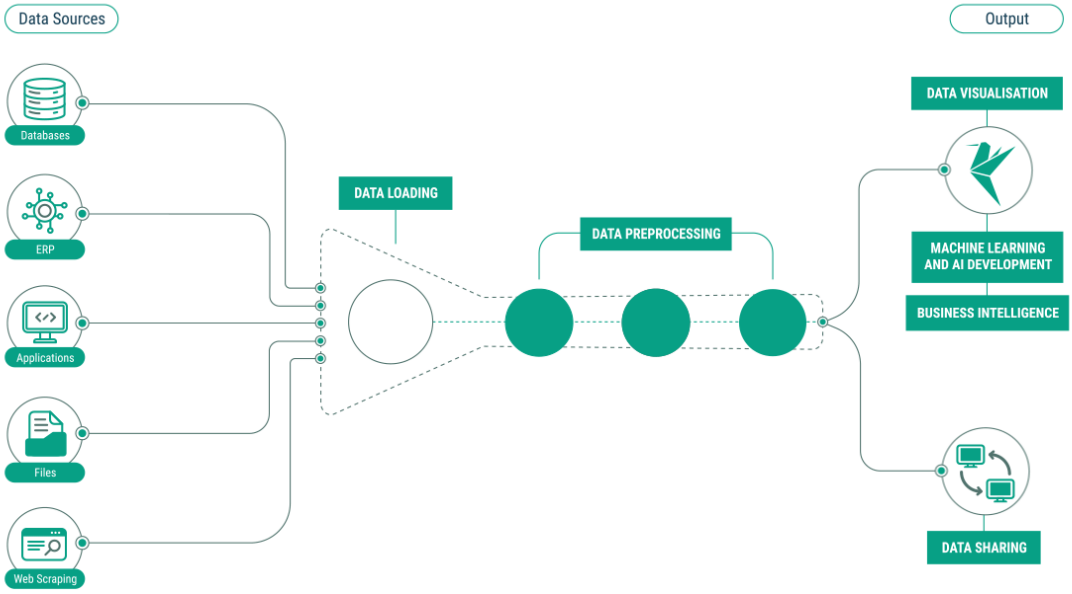


Slika 4.3 Koraki analize podatkov ML v R

Vir: Wickham et al. (2023).

Pri ML je običajno **zmotno prepričanje, da se večina časa in truda porabi za dejansko izdelavo algoritmov ML**. V resnici je ravno obratno, večina časa se običajno porabi za delo s podatki in naloge predobdelave, ne pa za postopek modeliranja. Včasih so vsi postopki pred postopkom modeliranja veliko bolj zahtevni in zahtevni. Zato ni soglasja o tem, kako je treba te korake izvesti. Na sliki 2.9 je predstavljen primer dobrega pristopa k preoblikovanju podatkov v poslovna spoznanja in splošno znanje. Postopek se začne s korakom uvoza, ki je eden najpomembnejših korakov pri gradnji modelov ML, saj brez uvoza podatkov v programsko opremo ni mogoče izvesti nobene analize. To običajno pomeni, da vzamete podatke, shranjene v datoteki, zbirki podatkov ali spletnem vmesniku za programiranje aplikacij (API), in jih naložite v podatkovni okvir v programu R (Wickham et al., 2023). Drugi korak je povezan z urejanjem podatkov, ki je postopek, edinstven za program R, in se nanaša na preoblikovanje podatkov v določeno obliko za nadaljnjo analizo (vsak stolpec je spremenljivka, vsaka vrstica pa je podatkovni okvir opazovanja-tibla). Naslednji korak je povezan s preoblikovanjem podatkov, ki običajno vključuje zožitev nabora opazovanj na podvzorec, ki nas zanima. Poleg tega lahko vključuje tudi ustvarjanje novih spremenljivk kot kombinacije več obstoječih ali generiranje zbirnih statistik. Vizualizacija in modeliranje imata na področju analize podatkov različni, vendar dopolnjujoči se vlogi. Vizualizacija je dejavnost, ki je v veliki meri osredotočena na človeka in ponuja vpogled, ki se lahko izmika bolj formaliziranim pristopom. Dobro pripravljena vizualizacija lahko razkrije nepričakovane vzorce, spodbudi nove poizvedbe in celo predlaga, da je prvotna vprašanja morda treba izboljšati ali uporabiti druge podatke. Nasprotno pa modeli zagotavljajo matematični ali računalniški okvir za odgovore na natančno oblikovana vprašanja. Ponujajo razširljivost in učinkovitost, zaradi česar so primerni za obdelavo velikih zbirk podatkov. Vendar imajo modeli (v katere so vključeni tudi ML) inherentne predpostavke in ne morejo postavljati vprašanj ali izpodbijati teh predpostavk. Posledično modeli morda ne

morejo presenetiti ali razkriti nepredvidenih spoznanj. Sinergija med vizualizacijo in modeliranjem se kaže v njuni skupni vlogi pri analizi podatkov. Vizualizacija pomaga pri začetnem raziskovanju in spodbuja oblikovanje natančnih vprašanj, medtem ko modeli sistematično zagotavljajo odgovore v okviru opredeljenih parametrov. Prepoznavanje prednosti in omejitev vsakega od pristopov je ključnega pomena, kar vodi k bolj celovitemu in informiranemu procesu analize podatkov. Zadnji korak predstavlja komunikacijo, ki je ključna za uspeh analize podatkov, saj če informacije niso posredovane nosilcu odločanja na pravilen in dosleden način, je lahko celotna analitika zaman. Ključni element podatkovne analitike so ML modeli, brez katerih ni mogoče izpeljati zaključkov o poslovnih procesih. Za reševanje specifičnih problemov SC je treba bolje prilagoditi arhitekturo za splošno namenske aplikacije poslovne inteligence (predstavljeno na sliki 4.2), prav tako tudi modele ML. V skladu s tem je podjetje **Equilibrium AI** za preoblikovanje delovanja oskrbovalnih verig s povečanjem operativne učinkovitosti, izboljšanjem odločanja in usmerjanjem k doseganju ciljev podjetja razvilo platformo AI & ML, predstavljeno na sliki 4.4.



Slika 4.4 Potek podatkov in znanja ML za podjetje Equilibrium AI

Vir: Equilibrium AI (n.d.).

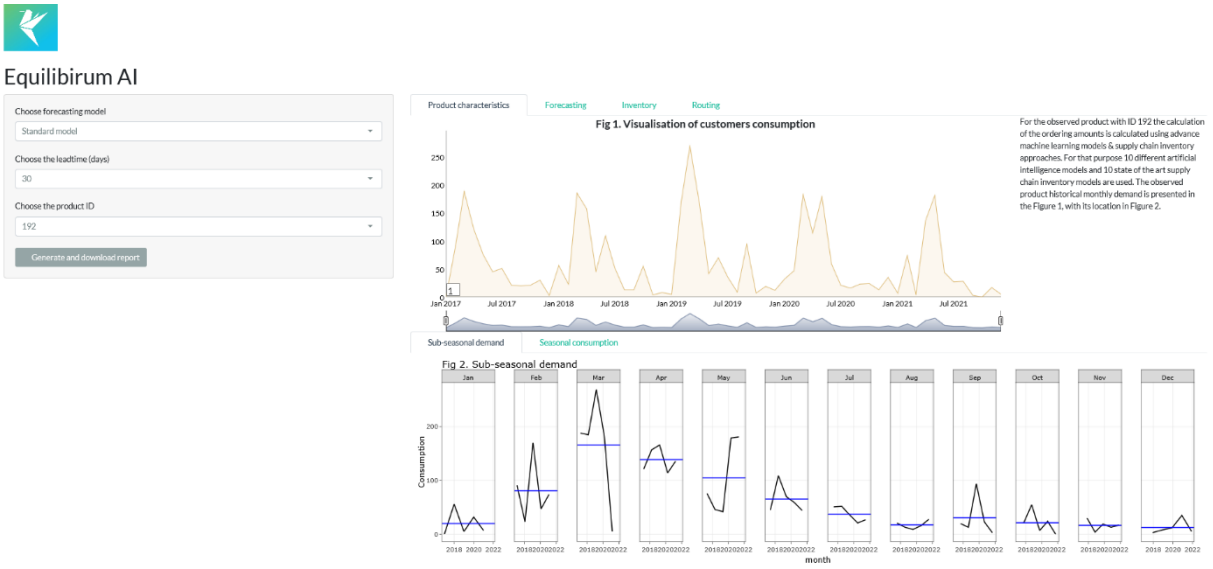
Slika predstavlja dober primer vsakodnevne prakse, kako se v aplikacijah SC pridobivajo znanja in vpogledi. Na splošno je postopek sestavljen iz operacij zaledja in sprednjega dela, da bi ustvarili vrednost (poslovna spoznanja) za uporabnike. Postopek backend se začne z ekstrakcijo podatkov iz različnih podatkovnih virov, ki jih običajno najdemo v SC:

- Podatkovne zbirke;
- sistemi za načrtovanje virov podjetja (SAP, Navigator, Microsoft Dynamics itd.);
- aplikacije (spletni vmesniki API);
- ploščate datoteke (csv, xlsx, JSON itd.);
- Splet, internet in drugi spletni viri.

Vsak vir podatkov ima drugačno strukturo, protokole in ustrezne postopke za pridobivanje in nalaganje podatkov za čiščenje in predobdelavo pred uporabo algoritmov ML. Zato se ta postopek izvaja s pomočjo nalagalnikov podatkov, ki imajo vnaprej programirano kodo za podatkovno rudarjenje različnih podatkovnih virov in prenos podatkov iz vrstic v novo podatkovno zbirko, ki je strukturirana in urejena za uporabo modelov ML. Pred uporabo modelov ML je potreben še en korak, ki se imenuje predobdelava podatkov. V tem koraku se podatki iz vrstic, zbrani od podjetij, preverijo glede na napačne vnose, nelogične vrednosti, pravilno strukturo vnosov, odstopanja, dvojne vnose, NA, NaN itd. Postopek se nadaljuje z združitvijo zunanjih podatkov s podatki podjetja. Ti podatki so običajno povezani z zunanjimi dejavniki, ki lahko potencialno vplivajo na opazovani poslovni proces SC, na primer vremenski podatki, indeks cen življenjskih potrebščin, povprečni dohodek v določeni regiji, posebne demografske značilnosti na določenem območju, cene plina, izbruhi pandemij, komentarji družbenih omrežij o izdelkih podjetja itd. To je zelo pomembno, ker se tako celostno zberejo vsi možni dejavniki (notranji in zunanji), ki bi lahko vplivali na določen poslovni proces, kar povečuje možnost, da bodo modeli ML v podatkih našli pravi signal in bodo lahko sprejeli pravilne sklepe in pravila, kateri so temeljni vzroki, zakaj se poslovni proces obnaša tako, kot je opazovan.

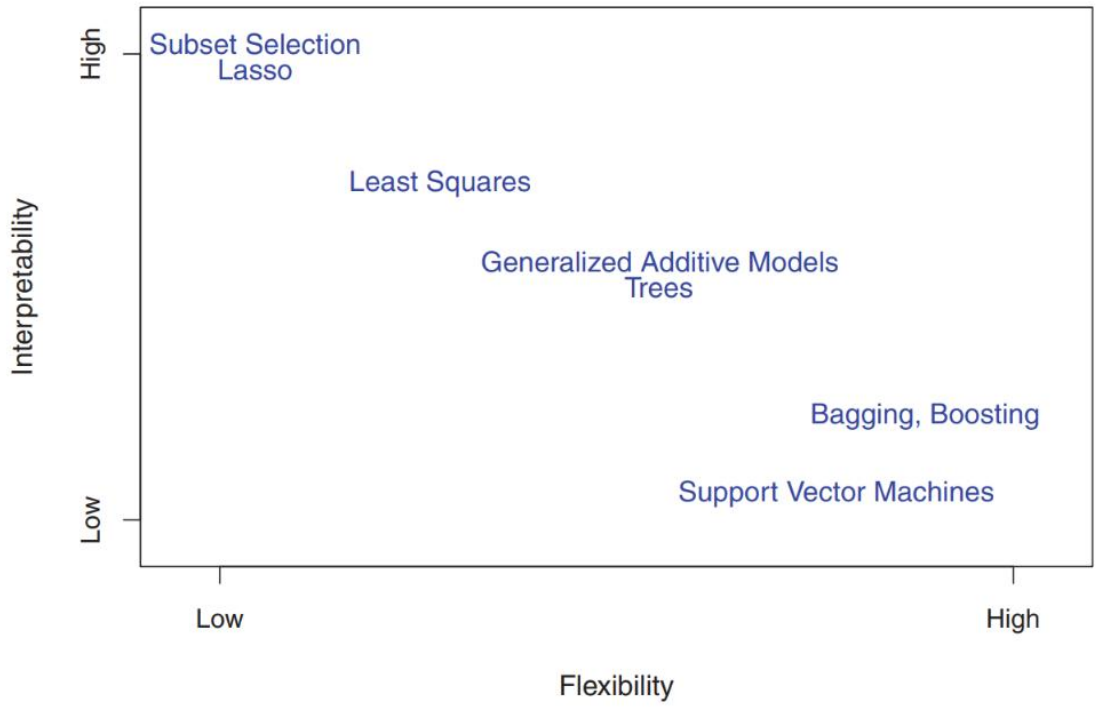
Po združitvi notranjih in zunanjih podatkov je predobdelava sestavljena iz zaznavanja signalov, odstranjevanja šuma iz podatkov, oblikovanja značilnosti in naključne razdelitve podatkov na učne in testne (včasih tudi na validacijske podatke, če je razvit model nevronske mreže). Podatki, pridobljeni v koraku predobdelave, se očistijo in strukturirajo za uporabo modelov ML.

Izhodni del je sestavljen iz vizualizacije podatkov, razvoja ML in AI ter izmenjave podatkov. Včasih se ti podatki delijo brez uporabe modelov ML z drugimi platformami, ki izvajajo različne vrste analiz (samo poročanje zainteresiranim stranem ali vladnim agencijam). Postopek vizualizacije se izvaja prek frontend dela platforme, ki je osredotočen na uporabnika in uporabnikom omogoča, da podajo zahteve o tem, katere podatke, kako in v kakšnih nastavitvah želijo videti opazovane podatke SC (na primer slika 4.5).



Slika 4.5 Tipični vizualizacijski del platforme ML v SC

Nasprotno pa je del ML za obdelavo podatkov skrit pred očmi uporabnikov in ga ni lahko razumeti. Zato se modeli ML včasih obravnavajo kot modeli **črne skrinjice**, pri katerih ni jasno, kako natančno je stroj povezal opazovani vhod z opazovanim izhodom. To je ena od ovir, ki preprečuje širšo uporabo modelov ML v praksi, zlasti tistih, ki so zapleteni za razlago (Rostami-Tabar in Mircetic, 2023). V skladu s tem bi lahko modele ML razdelili na tiste z visoko interpretabilnostjo - nizko prilagodljivostjo in nizko interpretabilnostjo - višjo prilagodljivostjo (slika 4.6). Na splošno velja, da se s povečevanjem prožnosti metode ML običajno povečuje natančnost modela ML in zmanjšuje interpretibilnost (Mirčetić et al., 2016).



Slika 46Prikaz kompromisa med prožnostjo in razlagalnostjo z uporabo različnih metod ML

Vir: Hastie et al. (2009).

4.3.1. Poslovni podatki ML in SC

Če uporabniki bolje razumejo vizualizacijo in grafiko, kot je slika 4.5, zakaj sploh potrebujemo ML in ali bi lahko preskočili modeliranje podatkov z ML ter naredili samo informativno grafiko? Žal ne. Morda je glavni razlog, zakaj potrebujemo modele ML, ta, da v vseh primerih ni mogoče imeti lahko berljivih in zaznavnih vzorcev v podatkih, vidnih prek grafike (kot na sliki 4.5). Pogostejša situacija je, da z grafiko običajno ne moremo razkriti skrivnosti dogajanja v opazovanih poslovnih podatkih SC in potrebujemo močnejša orodja v obliki algoritmov ML, da se poglobimo v podatke in poiščemo **pravila, ki ustvarjajo podatke** (slika 4.7).

Slika 4.7 Statistične značilnosti izdelkov v verigi preskrbe s hrano (povzeto za vse izdelke)

Na podlagi slike 4.7 je zelo težko preprosto sklepati in izpeljati poslovna pravila o postopku ustvarjanja podatkov. Za iskanje vzorcev v podatkih s slike smo razvili model ML, ki bi ga lahko uporabili za povzemanje značilnosti in odkrivanje pomembnih signalov v podatkih. V skladu s tem je razviti ML model za verigo preskrbe s hrano predstavljen v enačbi 1. Osnovno gonilo in hrbtenica tega modela ML je avtoregresijski integrirani model drsečega povprečja z naslednjo splošno obliko:

$$y_t' = c + (\phi_1 y_{t-1}' + \ldots + \phi_p y_{t-p}') + (\theta_1 e_{t-1} + \ldots + \theta_q e_{t-q}) + e_t; \quad (1)$$

$$y_t - y_{t-1} = c + \phi_1 (y_{t-1} - y_{t-2}) + \ldots + \phi_p (y_{t-p} - y_{t-p-1}) + (\theta_1 B e_t + \ldots + \underbrace{\theta_q e_{t-q}}_{\theta_q B^q e_t} + e_t);$$

$$y_t - B y_t = c + \phi_1 (y_{t-1} - B y_{t-1}) + \ldots + \phi_p (y_{t-p} - B y_{t-p}) + (e_t (1 + \theta_1 B + \ldots + \theta_q B^q));$$

$$(1 - B) y_t = c + \phi_1 (1 - B)(y_{t-1}) + \ldots + \phi_p (1 - B) y_{t-p} + e_t (1 + \theta_1 B + \ldots + \theta_q B^q);$$

$$(1 - B) y_t = c + \phi_1 (1 - B) B y_t + \ldots + \phi_p (1 - B) B^p y_t + e_t (1 + \theta_1 B + \ldots + \theta_q B^q);$$

$$\underbrace{(1 - B)^d y_t}_{\substack{differencing \\ d\_\deg ree}} \cdot \underbrace{(1 - \phi_1 B - \ldots - \phi_p B^p)}_{AR(p)} = c + \underbrace{e_t (1 + \theta_1 B + \ldots + \theta_q B^q)}_{MA(q)}.$$

Model ML jasno kaže na svojo nizko intepretabilnost in značilnosti črne skrinjice. Povprečen poslovni uporabnik težko razume povezave med vhodnimi in izhodnimi podatki. Poleg tega se povprečnemu poslovnemu uporabniku ob soočenju s predstavljenim modelom poraja

vprašanje! Kaj je enačba (1)? Lahko trdimo, da enačba (1) predstavlja pravila s slike 4.1, ki jih ustvari ML data & knowledge pipeline in ki razkrivajo skrivnost o procesih ustvarjanja podatkov v danem poslovnem okolju SC.

Na prvi pogled se zdi, da razviti model ML v enačbi (1) ne izboljša našega razumevanja podatkov. Še vedno smo zmedeni kot pri sliki 4.7, vendar ima ML model v primerjavi s sliko ključno prednost. V bistvu je model ML **matematična formula,** ki človeškemu uporabniku morda ni lahko razumljiva, je pa popolnoma razumljiva računalniku, ki ga **je mogoče programirati, da uporabi dano formulo** in **sprejme poslovne odločitve na** podlagi odkritih pravil.

VIRI

1. Chollet, F. (2021). Deep learning with Python. Simon and Schuster.
2. Equilibrium AI (n.d.). Equilibrium AI Data Pipeline [available: https://eqains.com/, access: January 23, 2024]
3. Hastie, T., Tibshirani, R., Friedman, J. H., & Friedman, J. H. (2009). The elements of statistical learning: Data mining, inference, and prediction (Vol. 2). Springer.
4. Jordan, M. I. & Mitchell, T. M. (2015). Machine learning: Trends, perspectives, and prospects. Science, 349(6245), pp. 255-260.
5. Mirčetić, D., Ralević, N., Nikoličić, S., Maslarić, M. & Stojanović, Đ. (2016). Expert system models for forecasting forklifts engagement in a warehouse loading operation: A case study. Promet-Traffic & Transportation, 28(4), pp. 393-401.
6. Ng, A. (2017). Machine learning yearning. [available: http://www.mlyearning.org/, access: January 23, 2024]
7. Rostami-Tabar, B. & Mircetic, D. (2023). Exploring the association between time series features and forecasting by temporal aggregation using machine learning. Neurocomputing, 548, 126376.
8. ScienceSoft (n.d.). Microsoft Business Intelligence to Drive Robust Analytics and Insightful Reporting [available: https://www.scnsoft.com/services/business-intelligence/microsoft, access: January 23, 2024]

9. Wickham, H., Çetinkaya-Rundel, M. & Grolemund, G. (2023). R for data science. O'Reilly Media, Inc.