

**Predtekmovanje.** IME DATOTEKE: predtekmovanje.py

Za predtekmovanje sem uporabil razred *LinearLearner* iz pripravljene datoteke *linear.py* za gradnjo modela z linearno regresijo. Pri predtekmovanju smo gradili napovedni model za samo eno linijo (in smer vožnje). Obdelavo vhodnih podatkov oz. oblikovanje značilk sem izvedel v funkciji *filter\_input()*, in sicer sem določil nove 0/1 značilke, s katerimi sem predstavil osnovne podatke kot npr. dan v tednu in ura začetka vožnje (ta dva podatka sta opazno najpomembnejša za določanje končnega časa vožnje). Oblikoval sem tudi dodatne 0/1 značilke z uporabo osnovnih podatkov, ki so npr. povedale kateri letni čas je, ali je praznik, ali so poletne počitnice (spremenjen vozni red). Pri gradnji značilk sem upošteval znanje o vožnji avtobusov, prometu in gibanju ljudi v mestu, ter testiral vpliv značilk na končno napoved z uporabo internega testiranja in ocene iz lestvice na strežniku.

**Tekmovanje.** Spodaj so opisane različne rešitve za tekmovanje, ki sem jih oddal na lestvico.

IME DATOTEKE: poly.py (rezultat: 203.39716)

V tej implementaciji sem poskušal napovedati trajanje vožnje s pomočjo polinomske regresije. Namesto množice 0/1 značilk sem attribute kot so ura normaliziral, da je bila njihova vrednost na zveznem intervalu  $[0, 1]$  in jih tako uporabil v polinomski regresiji. Model ni vračal dovolj dobrih rezultatov, zato sem ga opustil.

IME DATOTEKE: tekmovanje-17-12.py (rezultat: 196.80010)

Implementacija uporablja predlogo za linearno regresijo iz datoteke *linear.py* in 0/1 značilke, kot pri predtekmovanju. Ker gre za podatke o vseh avtobusnih linijah sem z uporabo razreda *SeparateBySetLearner* razdelil avtobusne linije na podlagi številke linije in smeri vožnje (npr. linija št. 6, smer Črnuče) in za vsako izmed njih zgradil ločen model. Kot dodatno značilko sem vpeljal podatke o vremenu (vir: ARSO), kjer sem upošteval količino dežja in količino no-vozapadlega snega v dnevu (funkciji *read\_weather()* in *weather\_marker()*).

IME DATOTEKE: tekmovanje\_FINAL.py\* [končna oddaja] (rezultat: 178.08593)

Glavna razlika v primerjavi z prejšnjo verzijo (*tekmovanje-17-12.py*) je v ustrezni obravnavi linij, za katere nisem imel že zgrajenega modela (ker se ta linija ni pojavila v učnih podatkih, zato njen model ni bil zgrajen) - taka je posebna linija 1 SEJEM. V tem primeru namesto povprečja napovedi vseh modelov uporabim kar "najbolj soroden" model, ki ga določim na podlagi enake številke linije. Tako sem omogočil bolj informirane in točne napovedi za linije, katere niso bile med učnimi podatki, ter tako precej izboljšal točnost.

**Izjava o izdelavi domače naloge.** Domačo nalogo in pripadajoče programe sem izdelal sam.