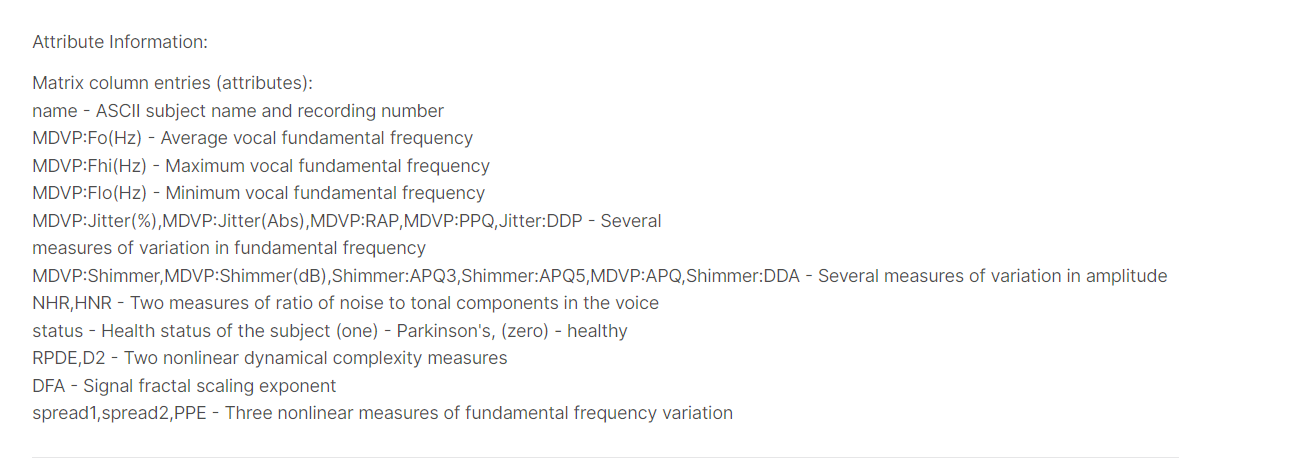
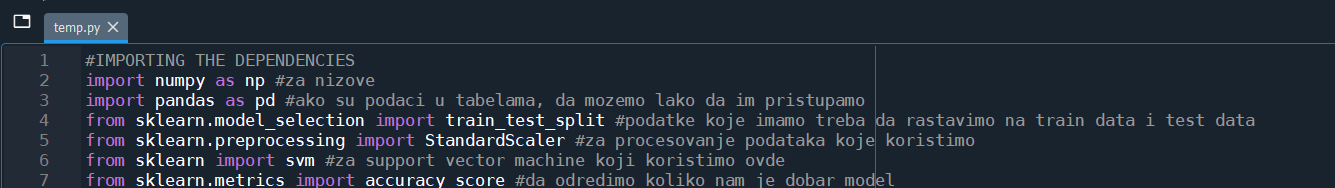
**DETEKCIJA PARKINSONOVE BOLESTI KORIŠĆENJEM MAŠINSKOG UČENJA**

Cilj ovog projekta je procena da li neka osoba ima Parkinsonovu bolest ili ne. Procena se vrši na osnovu baze podataka koja je smeštena u fajlu pod nazivom [**parkinsons**](file:///C:\Users\natal\Desktop\parkinsons.csv)**.csv**, koja sadrži vrednosti izmerenih sledećih parametara za veliki broj testiranih osoba. Na osnovu prethodnih iskustava sadržanih u bazi, procenjujemo da li trenutno testirana osoba ima Parkinsonovu bolest ili ne.

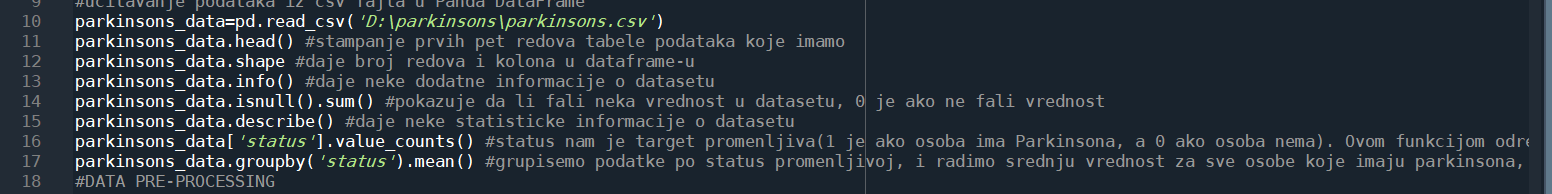


Ova baza je preuzeta sa sajta [**https://www.kaggle.com/datasets/nidaguler/parkinsons-data-set?resource=download**](https://www.kaggle.com/datasets/nidaguler/parkinsons-data-set?resource=download)**.**

Na slici su prikazane i opisane sve biblioteke koje smo koristili:



Neophodno je da podatke iz naše baze učitamo u panda data frame, kako bismo mogli da manipulišemo parametrima sadžanim u njoj, uz pomoć različitih funkcija. Pre svega, smeštamo ih u promenljivu pod nazivom parkinsons\_data. Neke od pomenutih funkcija su prikazane na sledećoj slici:

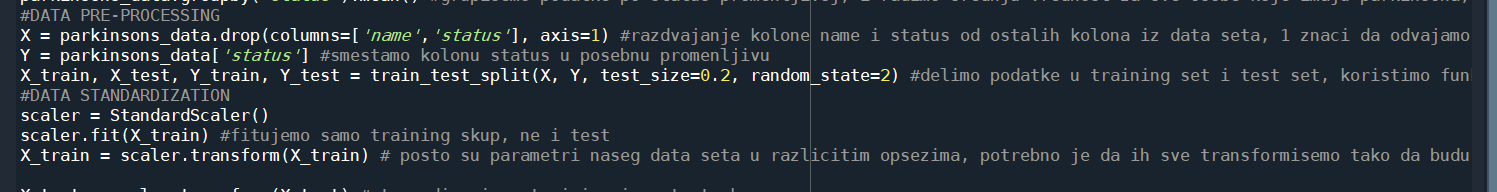


Target parametar naše baze je parametar pod nazivom **status**, i on može imati vrednosti 0 ili 1, koje ukazuju na to da li osoba ima Parkinsonovu bolest ili ne.

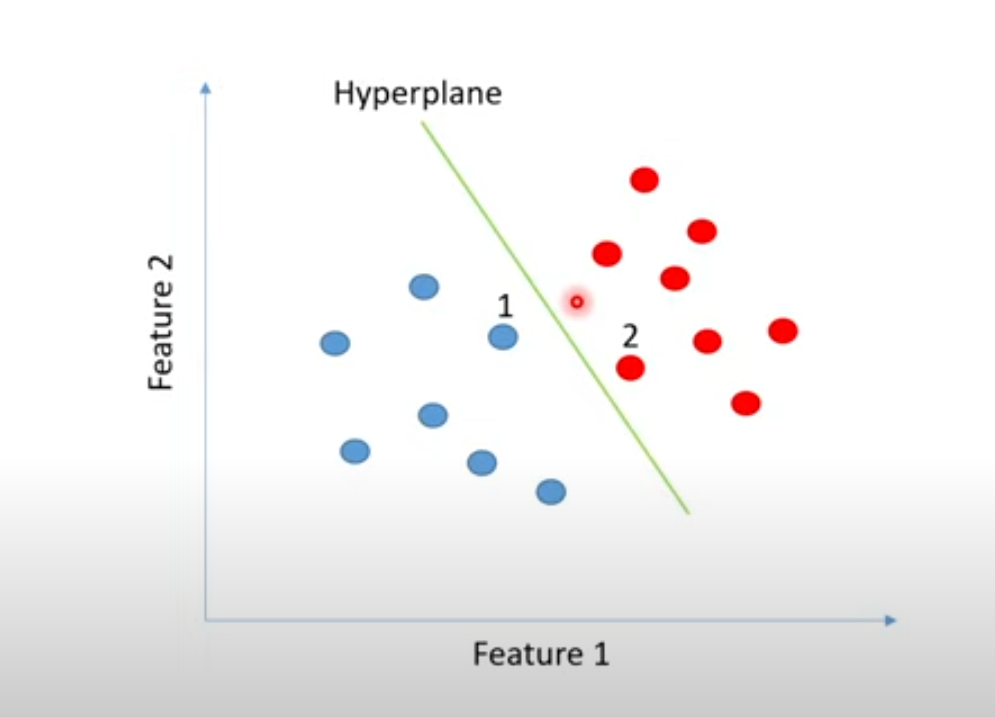
1. Osoba ima Parkinsonovu bolest
2. Osoba nema Parkinsonovu bolest

Pošto je cilj da naš program vrati vrednost ovog parametra, tj. status parametar smatramo target parametrom, neophodno je da kolonu sa vrednostima statusa odvojimo od ostalih kolona baze.

Takođe, razdvajamo podatke na training set i test set. Na training setu ćemo trenirati naš model, a na test setu ćemo proveravati da li model radi na pravilan način.

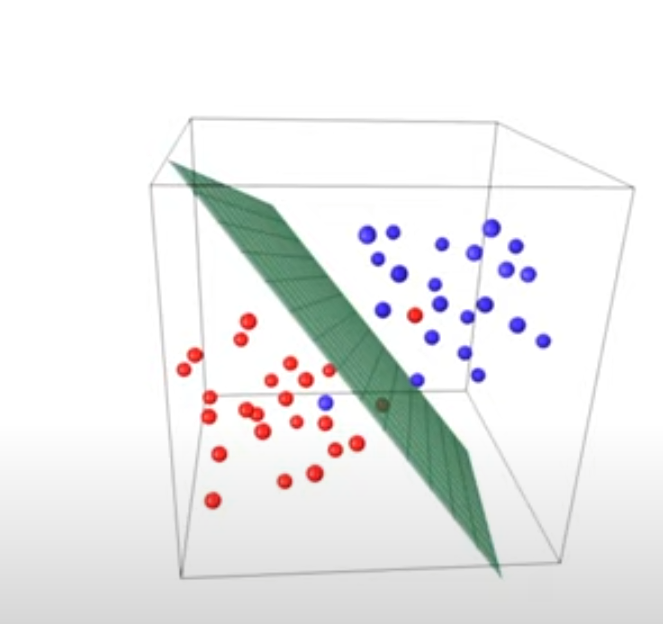


Nakon što smo se pozabavili uređivanjem podataka, sledeći korak je treniranje modela. Za treniranje koristimo metodu pod nazivom **Support-vector machine algoritam.** Potrebno je pronaći hyperplane (na slikama to je linija ili ravan), koja razdvaja sve podatke na neke dve grupe. U našem slučaju to su grupe: oboleli i zdravi.



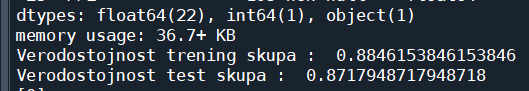
 Ovo su tačke koje su najbliže hyperplane liniji.

Kada pomeramo hyperplane, pomeraju se i support vectors. Takođe, kada dodamo nove podatke u model, koji je već istreniran, on pokušava da rasporedi te podatke sa određene strane hyperplane, tj. da ih smesti u određenu grupu. Slika iznad je prikaz 2d modela, tj. situacije u kojoj imamo samo dva parametra na osnovu kojih raspoređujemo podatke. Sa povećanjem broja parametara, povećava se i dimenzija modela. Npr. na narednoj slici je prikazan 3d model.

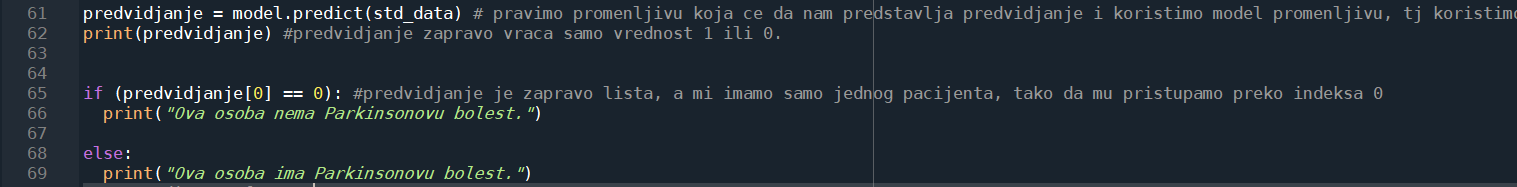


Pošto naš model ima vrlo malo podataka, treniranje se vrši jako brzo.

Nakon što smo istrenirali skup, potrebno je da odredimo verodostojnost modela. Radimo to i za jedan i za drugi skup. Da bismo bili sigurni da model dobro radi, potrebno je da verodostojnost i jednog i drugog skupa bude preko 75%, pri čemu razlika između verodostojnosti training i test skupa mora da bude minimalna. U našem slučaju su ovi uslovi ispunjeni:



Krajnji korak je pravljenje sistema predviđanja. Možemo isprobati da li model radi, tako što ćemo iz baze preuzeti parametre vezane za jednu osobu. Na sledeći način primenjujemo naš model na unete parametre. Koristeći funkciju predict, pravimo promenljivu predviđanje, koja će samo sadržati vrednost status parametra, tj. jedinicu ili nulu.



Ukoliko je vrednost promenljive prediviđanje jednaka jedinici, program će štampati : „Ova osoba ima Parkinsonovu bolest“, a u suprotnom „Ova osoba nema Parkinsonovu bolest“.

Primenjujući ovaj postupak na više osoba iz baze, možemo zaključiti da naš model u svakom slučaju vraća ispravnu vrednost status parametra, odnosno na pravilan način predviđa da li osoba ima bolest ili ne. Zbog toga zaključujemo da je model dobro istreniran.



Naš projekat je rađen u programu **Spider**, a kod je smešten u fajlu <file:///D:\parkinsons\parkinsonova.py>. Pokretanje se vrši jednostavnim klikom na dugme F5 na tastaturi.

**Natalija Andrić 620/2019, Ivana Đokanović 603/2019**