Zadanie 1

Načítanie dát

Dáta sa nachádzali v csv súbore. Načítal som ich pomocou knižnice pandas do takzvaného dataframu.

Predspracovanie dát

Prvým krokom zistenie či sa v datasete nenachádzajú miesta bez hodnotu. Našiel som dve také: konrétne stĺpec active a ap_lo. Kedže išlo len o dve hodnoty v celom datasate dovolili som si ich doplniť priemernou hodnotou.

Ďalším krokom bolo nahradenie slovných výrazov číslami. Konkrétne išlo o stĺpce gender, cholesterol a glucose. Slová som nahradil nasledovne:

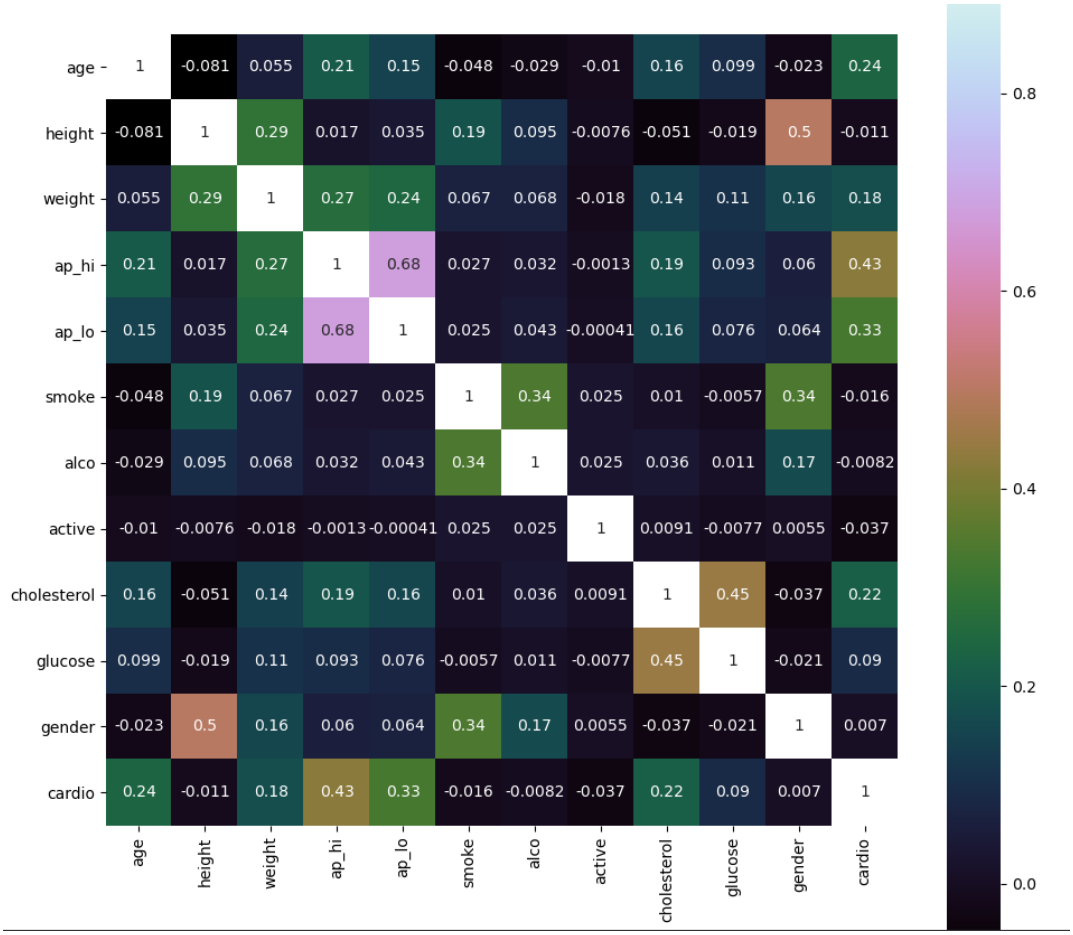
- Normal – 0
- Above normal – 1
- Well above normal – 2

Ako tretí krok som sa pozeral na grafy. Pomocou vykreslených grafov som sa z datasetu snažil vylúčiť nesprávne/nezmyselné hodnoty. Ako príklad možem uviesť vek. Vylúčil som všetky vzorky (riadky z datasetu) , kde vek bol väčší ako 100 rokov. Obdobne som postupoval aj pri ďaších stĺpcoch.

Nakoniec som sa ešte pokúsil stĺpce znormalizovať. Teda, konkrétne len stĺpec vek. Keďže bol zadávaný v dňoch boli to obrovské čísla. Predelil som tieto hodnoty číslom 365. Dostal som teda roky.

Analýza príznakov

Čo je najmocnejší ukazovateľ na prítomnosť srdcovej choroby? Na zodpovedanie tejto otázky som využil korelačnú maticu. Z nej vyplynulo že medzi najsilnejších ukazovateľ patrí tlak, vek, cholesterol, váha a glukóza. Ostatné parametre boli zanedbateľné. Preto som ich z ďalšej analýzy vylúčili. Vylučil som taktiež aj parameter ap_lo, a to z dôvodu že úzko súvisí s parametrom ap_hi.

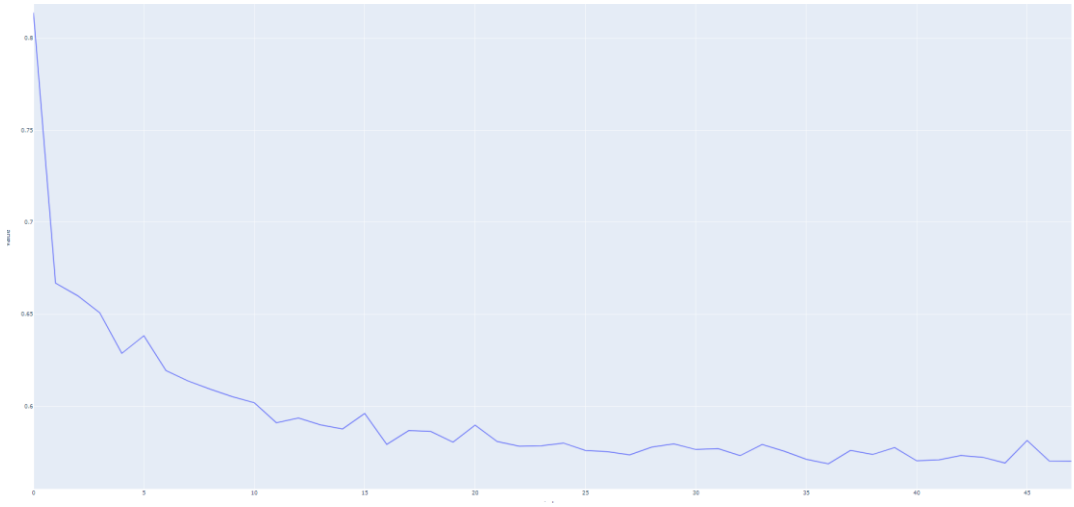


Binárny klasifikátor

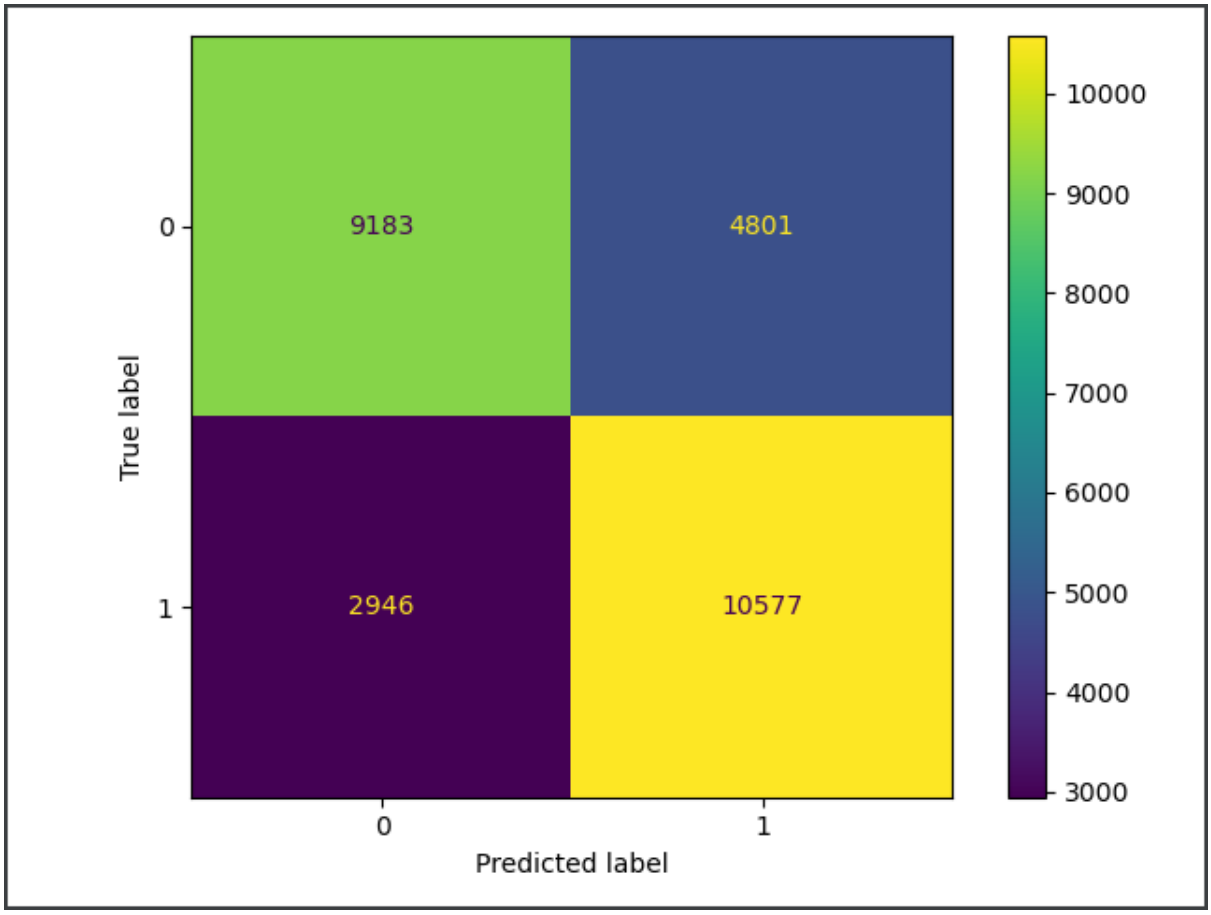
Prvým krokom bolo rozdelenie dát na trénovacie a testovacie. Ako klasifikátor som vybrali MLPClassifier z knižnice sklearn.neural_network. O nastavení jednotlivých parametrov som sa rozhodaval na základe ich popisu:

https://scikit-learn.org/stable/modules/generated/sklearn.neural_network.MLPClassifier.html

Vybraný solver bol „adam“. So 100 neurónmi na skrytej vrstve. Ako learning rate som zvolil hodnotu 0.01. Pri tejto hodnote loss krivka vyzerala najlepšie. Pre minimálne zlepšenie bola zvolená hodnota 0.00000001.

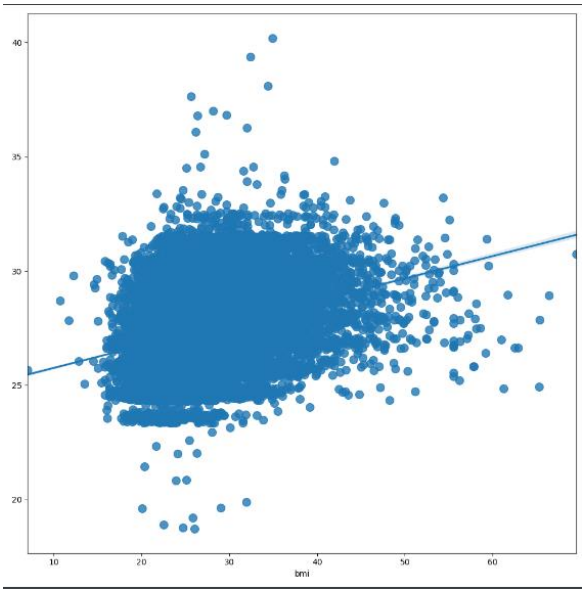


Vyskúšal som vela iných variácií nastavenia parametrov. Avšak sieť s takýmito parametrami dosahovala najlepšie výsledky. Úspešnosť siete bola stabilných 70% na testovacích dátach. To je možné vidieť aj takzvanom confusion matrix-e nižšie:



Regresia

Prvou bodom bol výpočet BMI. Aby som mali voči čomu sieť trénovať. Po vykreslení grafu s BMI som prišiel na to že niektoré údaje sú mimo realitu, preto som ich vyhodil z datasetu. Taktiež som z datasetu vyhodil stĺpce ktoré podľa korelačnej matice nemali veľký vplyv na výsledné BMI. A samozrejme aj stĺpce výška a váha keďže práve z tých dvoch stĺpcov bol vyrátaný stĺpec BMI.



Nasledovalo rozdelenie dát na trénovaciu a testovaciu množinu. Po vyrátaní regresie pomocou metódy najmenších štvorov a predpovedaní na testovaích dátach som dostal nasledujúce výsledky:

mse = 24.9

r2 = 0.09

Ďalším bodom bolo trénovanie neurónovej siete. Do neurónovej siete som dal tie isté dáta čo aj do lineárneho regresora. Pre neurónovú site som použil nasledujúce parametre:

- Solver: sgd
- Počet neurónov v skrytej vrstve: 100
- Activačná funkcia: logistic
- Learning rate: invscaling
- Minimálna chyba: tol=0.00000001
- Maximálny počet iterácii: 1000

V skutočnosti som ich vyskúšal omnoho viac, takto však dávali najlepšie výsledky. Po natrénovaní neurónovej siete a následnom predikovaní na testovacích dátach som dostal nasledujúce výsledky:

mse = 27.5

r2 = 0.01

Kedže vo väčšine prípadov platí čím menšia chyba (mse) tým lepšie a zároveň r2 chceme čo najväčšie, porovnaním som zistil že prvý model je lepší.