SLOVENSKÁ TECHNICKÁ UNIVERZITA V BRATISLAVE

FAKULTA INFORMATIKY A INFORMAČNÝCH TECHNOLÓGIÍ

Zadanie 4

KLASIFIKÁCIA BODOV V 2D PRIESTORE

Anna Yuová

Predmet: Umelá inteligencia

Akademický rok: 2020/2021

Semester: zimný

ZADANIE ÚLOHY

Máme 2D priestor, v ktorom sa nachádzajú súradnice X,Y od -5000 do 5000. Každý bod má unikátne súradnice, čiže nie je viac rovnakých bodov na tom istom mieste. Každý bod je zaradený do jednej zo štyroch tried: červená, zelená, modrá fialová. Na začiatku je v priestore týchto 20 bodov:
R: [−4500, −4400], [−4100, −3000], [−1800, −2400], [−2500, −3400] a [−2000, −1400]
G: [+4500, −4400], [+4100, −3000], [+1800, −2400], [+2500, −3400] a [+2000, −1400]
B: [−4500, +4400], [−4100, +3000], [−1800, +2400], [−2500, +3400] a [−2000, +1400]
P: [+4500, +4400], [+4100, +3000], [+1800, +2400], [+2500, +3400] a [+2000, +1400]

Úlohou je naprogramovať klasifikátor pre nové body vo funkcii classify(int X, int Y, int k), ktorá obsahuje súradnice nového bodu X a Y. Tento bod pridá do 2D priestoru a vráti triedu, ktorú pridelila pre tento bod. Na klasifikáciu treba použiť k-NN algoritmus, pričom k je 1, 3, 7 alebo 15.

Na demonštráciu klasifikátora treba vytvoriť testovacie prostredie, v ktorom sa budú postupne generovať nové body a funkcia classify ich bude prideľovať triedam. Celkovo treba vygenerovať 20 000 bodov (5 000 v každej triede). Súradnice nových bodov sa generujú náhodne a nový bod musí mať inú triedu (dva generované body za sebou nebudú patriť do tej istej triedy).

R body by mali byť generované s 99% pravdepodobnosťou s X < +500 a Y < +500
G body by mali byť generované s 99% pravdepodobnosťou s X > −500 a Y < +500
B body by mali byť generované s 99% pravdepodobnosťou s X < +500 a Y > −500
P body by mali byť generované s 99% pravdepodobnosťou s X > −500 a Y > −500

Návratovú hodnotu funkcie classify treba porovnať s vygenerovaným bodom a zistiť úspešnosť klasikátora.

OPIS RIEŠENIA

Na začiatku som si vytvorila dve polia – pre x-ové a y-ové súradnice počiatočných 20 bodov. Prvých 20 bodov som si zvolila podľa zadania – tie isté ako boli zadané. Vytvorila som si triedu Bod, do ktorej som postupne posielala x-ové a y-ové súradnice a vytvorila som si tak bod, ktorý som pridelila do 2D priestoru. Ďalej som prvým 5 bodom pridelila červenú triedu, ďalším 5 zelenú triedu, ďalším 5 modrú triedu a posledným 5 bodom fialovú triedu.

Vo for cykle generujem postupne 20 000 bodov, ktoré idem postupne klasifikovať a pridávať do priestoru. Vo vnútri for cyklu mám vnorený nekonečný while cyklus, ktorým ošetrujem to, aby som 2 bodom po sebe dvakrát nepridelila tú istú triedu. Druhým nekonečným while cyklom ošetrujem generovanie bodov s 99% pravdepodobnosťou, že budú patriť do nejakej triedy. Z každých 100 bodov by teda 1 nemal byť vygenerovaný z intervalov, ktoré očakávam pre danú triedu ale generujem ho z celého priestoru. Body sa snažím generovať tak, aby triedy, ktoré predpokladám, že by mali dostať, išli postupne za sebou - červená, zelená, modrá, fialová. Ak náhodou klasifikátor rozhodne prideliť dvakrát po sebe tú istú triedu, musím bod vygenerovať znovu a znovu ho poslať klasifikátoru, až kým nepridelí novú triedu (odlišnú od predošlej). Pri vygenerovaní bodu očakávam, že bude na 99% z danej triedy a očakávanú triedu si poznačím. Potom bod pošlem klasifikátoru, v ktorom prebieha k-NN algoritmus vo funkcii classify.

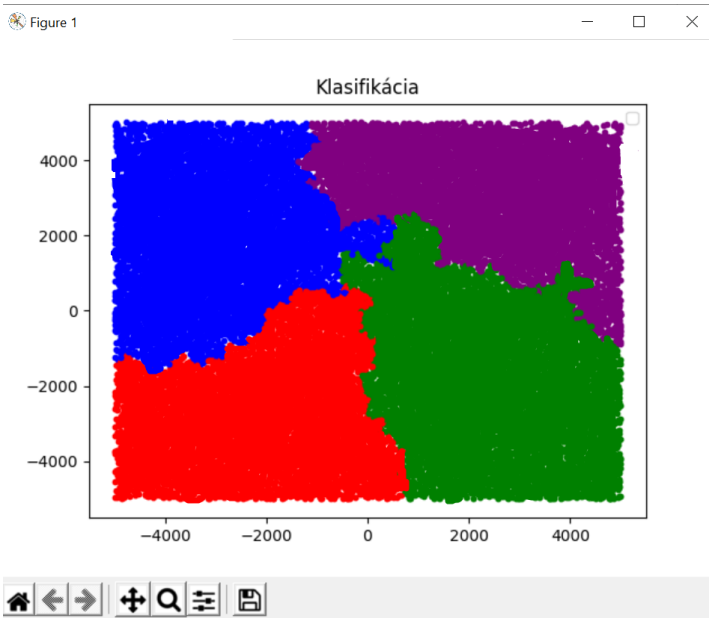
Do funkcie classify si pošlem x-ové a y-ové súradnice bodov a hodnotu k. Najprv si vypočítam vzdialenosť môjho bodu od všetkých bodov, ktoré aktuálne v priestore sú pomocou euklidovskej vzdialenosti. Vo funkcii euklidovska_vzdialenost si vypočítam rozdiel x-ových a y-ových súradníc a pomocou pytagorovej vety ich dám na druhú, spočítam a to celé odmocním (výsledok = odmocnina z x na druhú + y na druhú, kde x a y sú už rozdiely oboch súradníc). Tento výsledok si vo funkcii classify ukladám postupne do pola vzdialeností ako tuple – ukladám si vzdialenosti a farbu. Následne si toto pole vzdialeností zoradím pomocou funkcie sort od najmenšej vzdialenosti po najväčšiu. Podľa čísla k, vyberiem prvých k prvkov z tohto poľa a podľa toho viem, že toto sú napr. 3 najbližšie body k môjmu bodu, ak je k 3. Z tuple si urobím pole a vypočítam si z tých k vybraných bodov, koľko farieb z každej sa tam nachádza a vyberiem tú farbu, ktorá m najväčší počet. Toto robím v prípade, že je k rôzne od 1. Ak je k 1, tak vyberám rovno najbližší bod, ktorý sa k môjmu bodu nachádza.

Funkcia classify mi teraz vrátila triedu, do ktorej môj bod zaradila. V triede bod mám funkciu nastavFarbu, do ktorej pošlem túto farbu a pridelím ju danému bodu. Ak sa táto trieda zhoduje s triedou, ktorú som očakávala, že tam bod zaradí, zvýšim si počítadlo dobrých, ak sa nezhodujú, zvýšim počítadlo zlých. Na konci vydelím počítadlo zlých s počtom všetkých bodov, aby som zistila celkovú úspešnosť môjho klasifikátora. Potom už len zavolám funkciu vykresliBody, do ktorej si pošlem pole všetkých bodov, ktoré sú v mojom priestore a cez funkciu plt.scatter a plt.show ich vykresľujem.

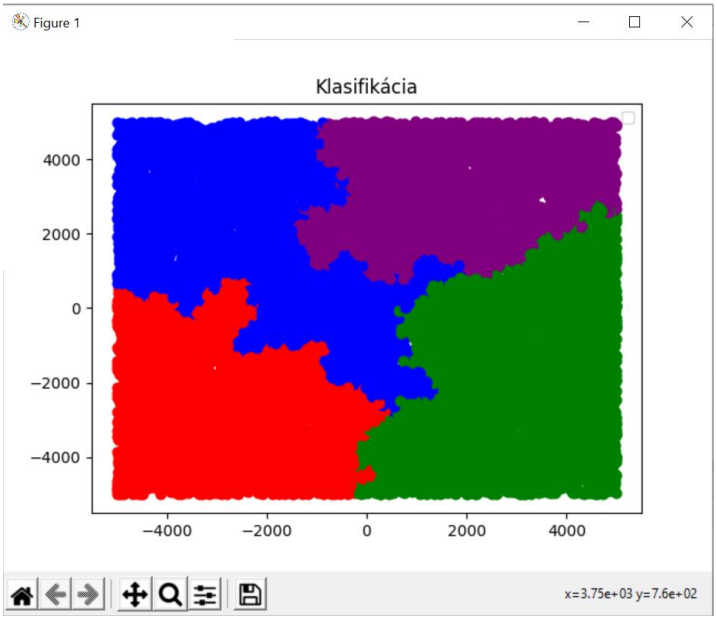
TESTOVANIE

Na otestovanie som si vygenerovala náhodné body do pola (okrem počiatočných, tie sú stále tie isté) a použila som ich pre všetky k = 1, 3, 7, 15 a počet bodov 20 000. Pri ďalšom testovaní som si vygenerovala nové body a znovu som použila tie isté body pre všetky 4 rôzne k . V programe som menila iba počet bodov a k.

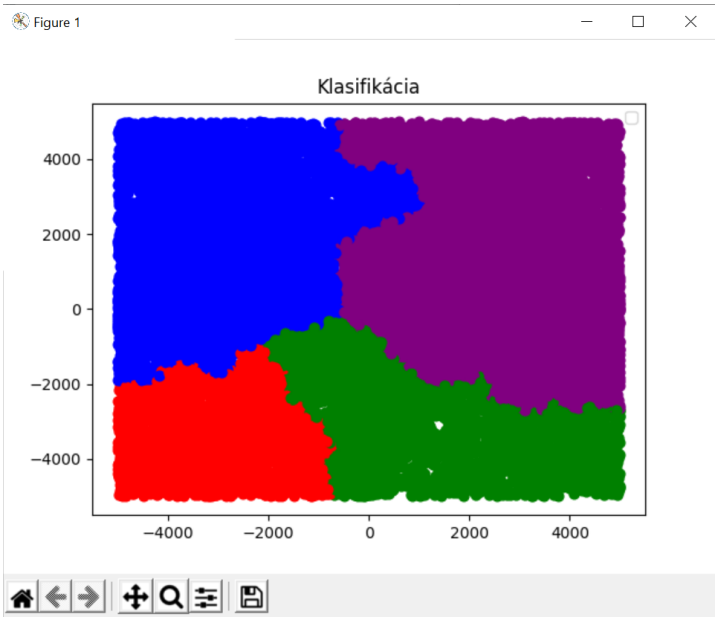
1. počet bodov: 20 000
   k = 1
   čas trvania : 31 minút
   rozdiel očakávanej triedy od triedy, ktorú určil klasifikátor: 4485 určil zle
   úspešnosť klasifikátora = 78%



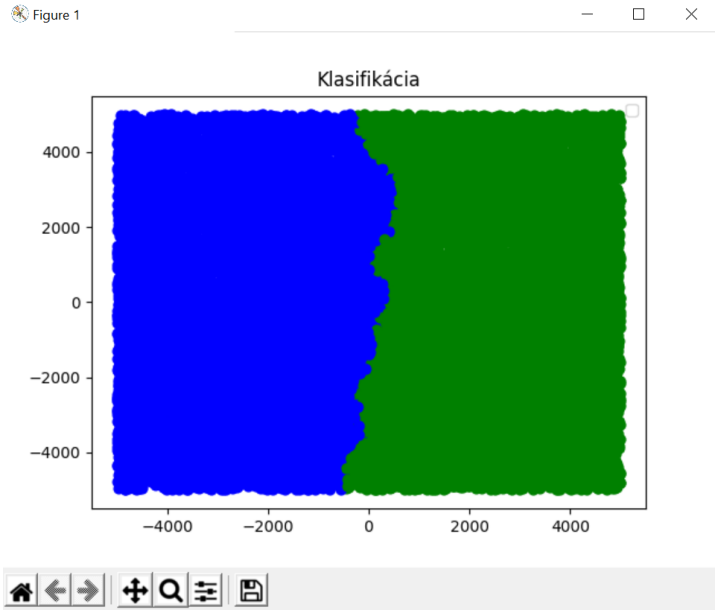
2. počet bodov: 20 000
   k = 3
   čas trvania : 22 minút
   rozdiel očakávanej triedy od triedy, ktorú určil klasifikátor: 5409 určil zle
   úspešnosť klasifikátora = 73%



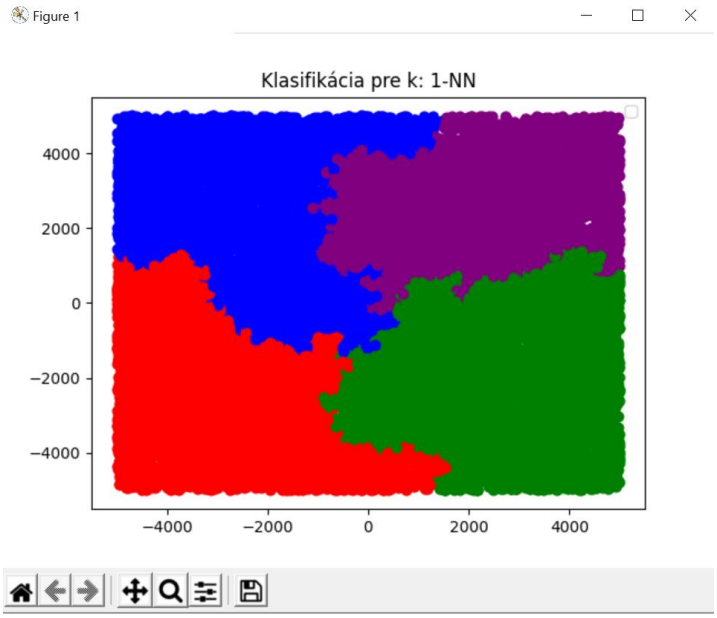
3. počet bodov: 20 000
   k = 7
   čas trvania : 21 minút
   rozdiel očakávanej triedy od triedy, ktorú určil klasifikátor: 6211 určil zle
   úspešnosť klasifikátora = 69%



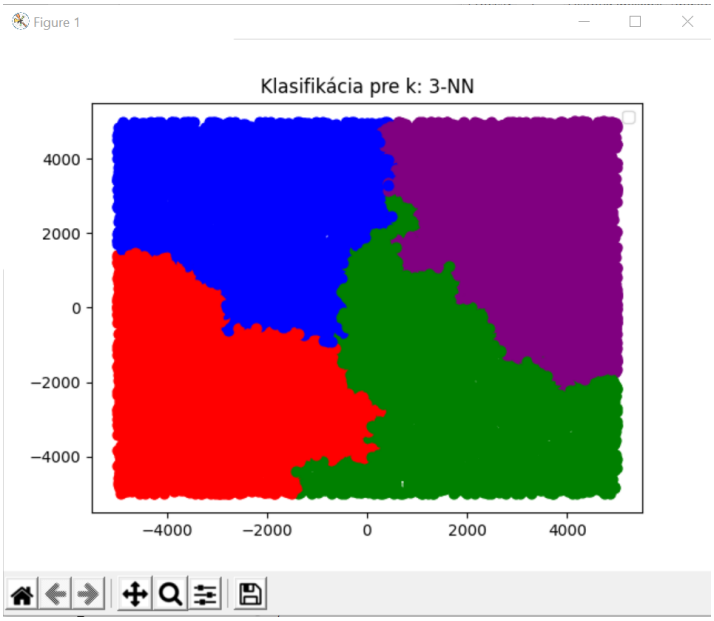
4. počet bodov: 20 000
   k = 15
   čas trvania : 18 minút
   rozdiel očakávanej triedy od triedy, ktorú určil klasifikátor: 10 948 určil zle
   úspešnosť klasifikátora = 46%



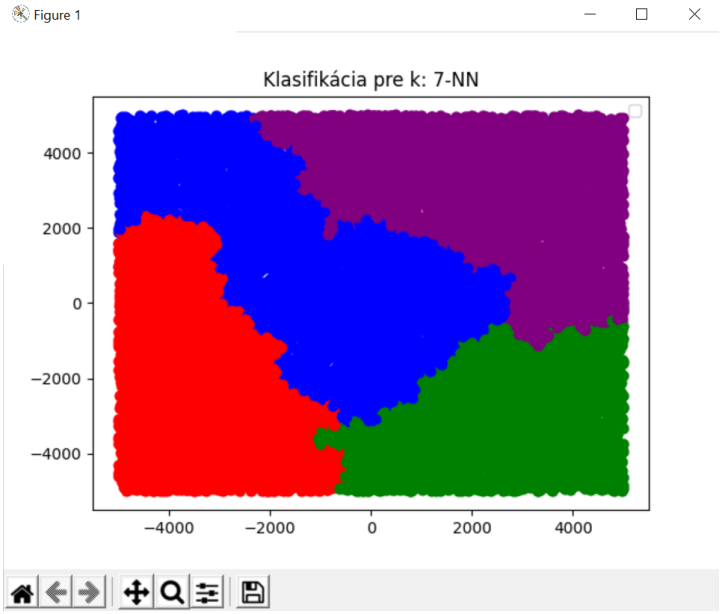
5. počet bodov: 20 000
   k = 1
   čas trvania : 21 minút
   rozdiel očakávanej triedy od triedy, ktorú určil klasifikátor: 5018 určil zle
   úspešnosť klasifikátora = 80 %



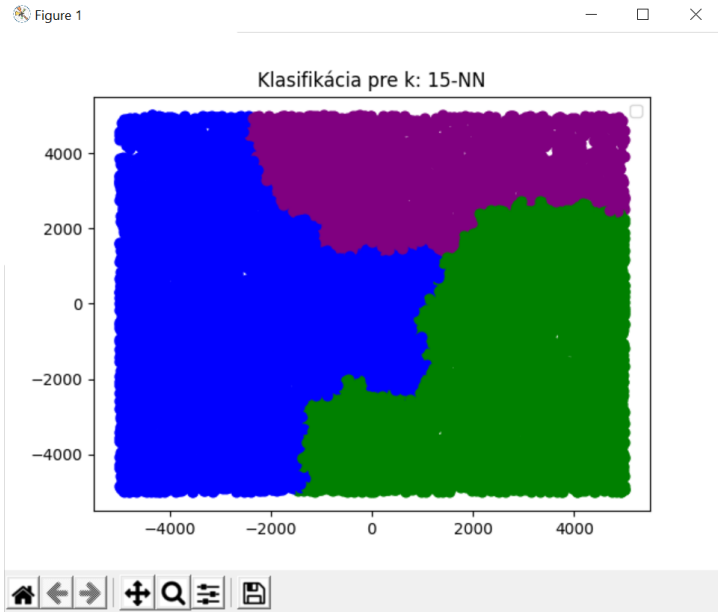
6. počet bodov: 20 000
   k = 3
   čas trvania : 22 minút
   rozdiel očakávanej triedy od triedy, ktorú určil klasifikátor: 4537 určil zle
   úspešnosť klasifikátora = 78 %



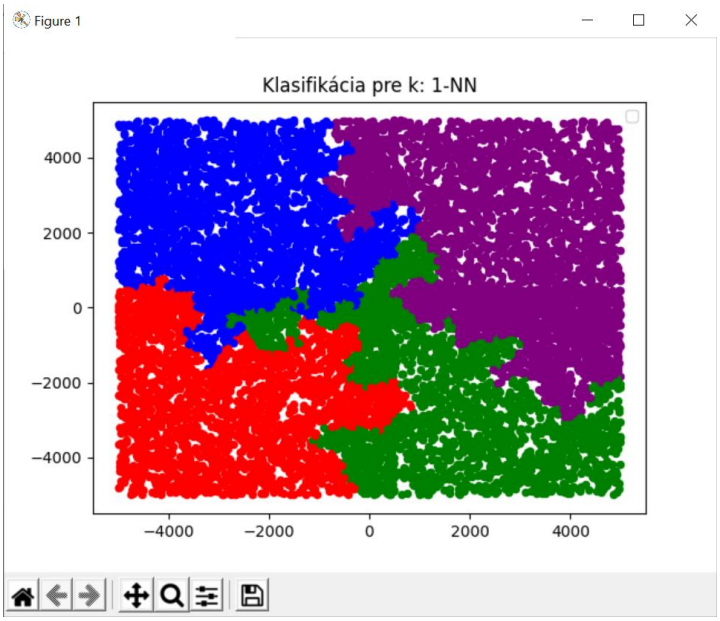
7. počet bodov: 20 000
   k = 7
   čas trvania : 23 minút
   rozdiel očakávanej triedy od triedy, ktorú určil klasifikátor: 6012 určil zle
   úspešnosť klasifikátora = 70 %



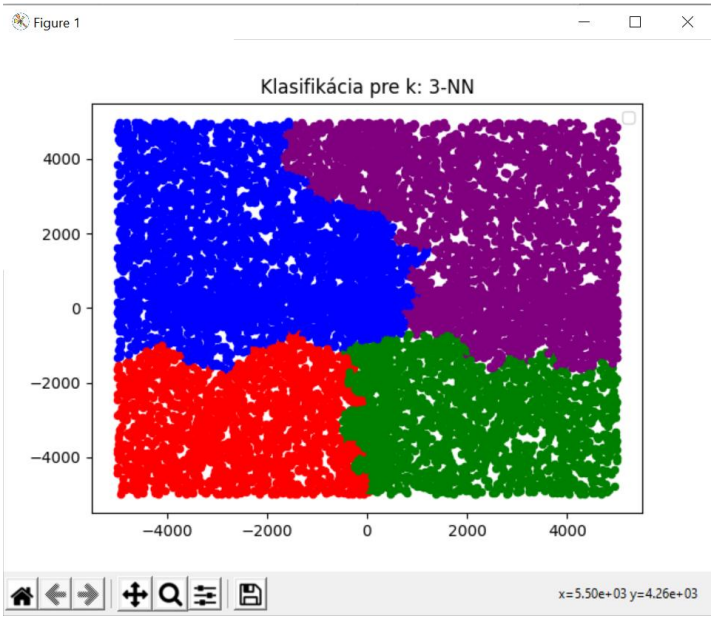
8. počet bodov: 20 000
   k = 15
   čas trvania : 23 minút
   rozdiel očakávanej triedy od triedy, ktorú určil klasifikátor: 9 312 určil zle
   úspešnosť klasifikátora = 54 %



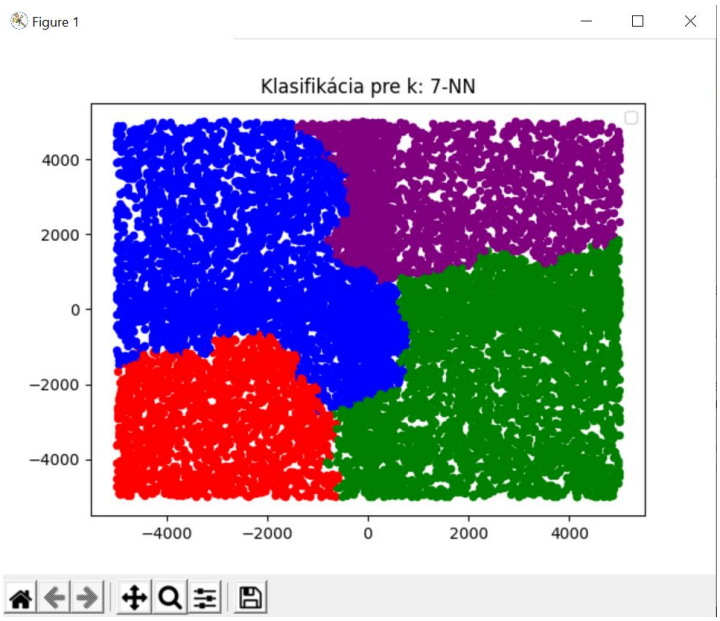
9. počet bodov: 11 000
   k = 1
   čas trvania : 8 minút
   rozdiel očakávanej triedy od triedy, ktorú určil klasifikátor: 2 258 určil zle
   úspešnosť klasifikátora = 80 %



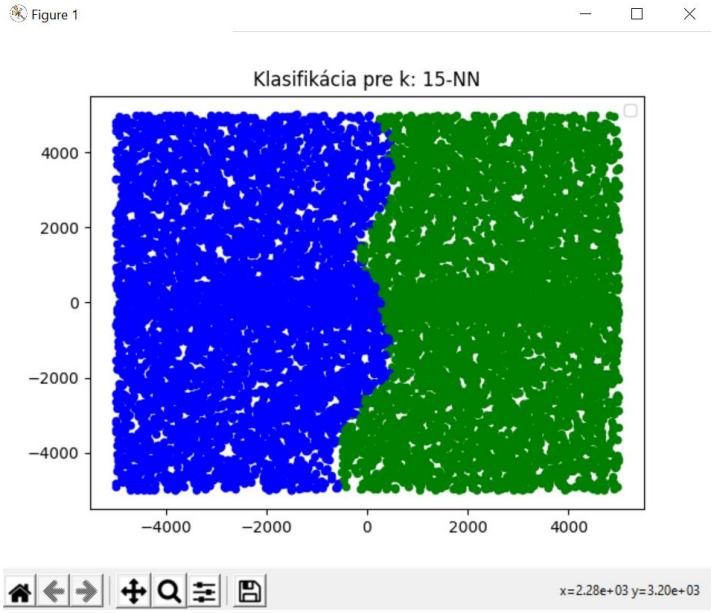
10. počet bodov: 11 000
    k = 3
    čas trvania : 9 minút
    rozdiel očakávanej triedy od triedy, ktorú určil klasifikátor: 2 564určil zle
    úspešnosť klasifikátora = 77 %



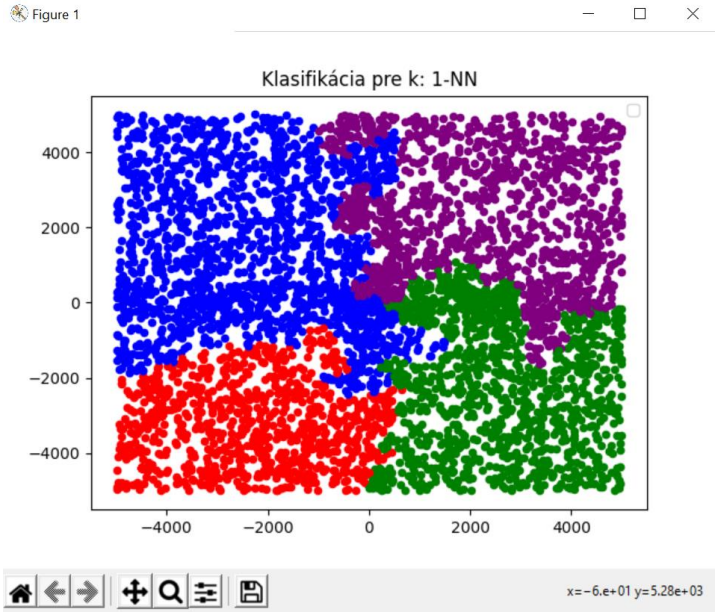
11. počet bodov: 11 000
    k = 7
    čas trvania : 10 minút
    rozdiel očakávanej triedy od triedy, ktorú určil klasifikátor: 2 994 určil zle
    úspešnosť klasifikátora = 73 %



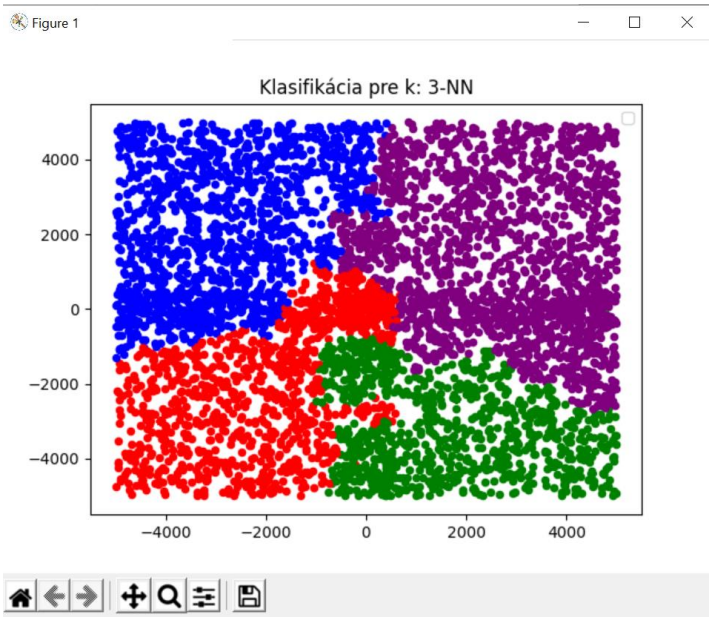
12. počet bodov: 11 000
    k = 15
    čas trvania : 10 minút
    rozdiel očakávanej triedy od triedy, ktorú určil klasifikátor: 5 500 určil zle
    úspešnosť klasifikátora = 50 %



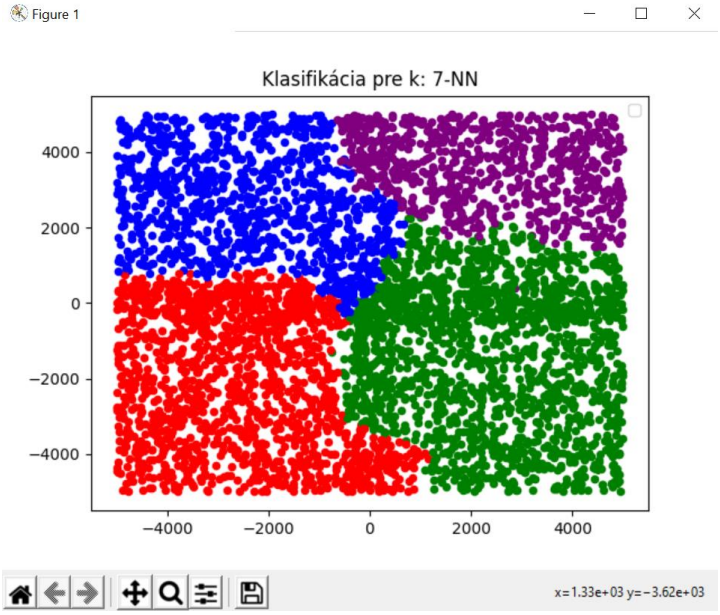
13. počet bodov: 5 000
    k = 1
    čas trvania : 2 minúty
    rozdiel očakávanej triedy od triedy, ktorú určil klasifikátor: 1 282 určil zle
    úspešnosť klasifikátora = 75 %



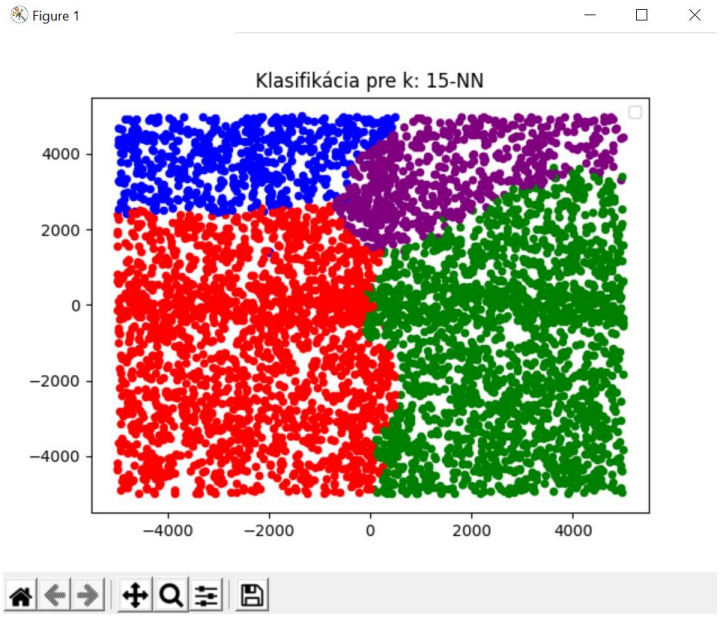
14. počet bodov: 5 000
    k = 3
    čas trvania : 2 minúty
    rozdiel očakávanej triedy od triedy, ktorú určil klasifikátor: 1 408 určil zle
    úspešnosť klasifikátora = 71 %



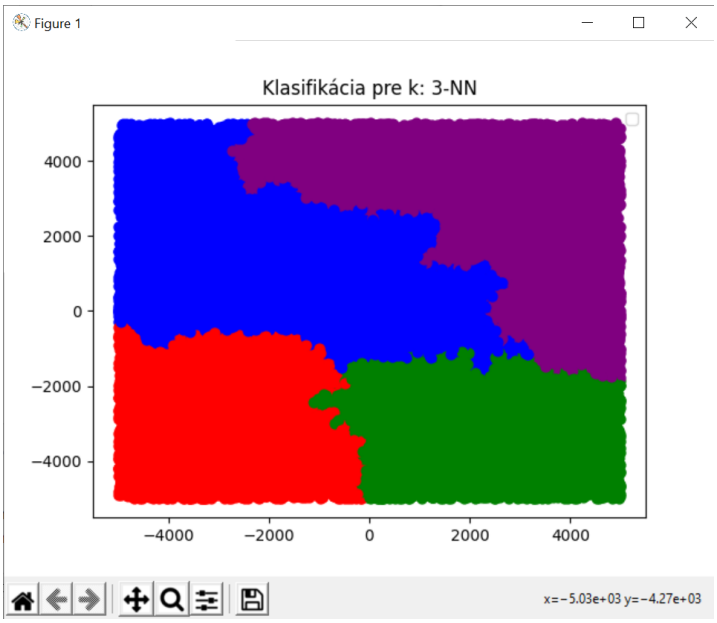
15. počet bodov: 5 000
    k = 7
    čas trvania : 2 minúty
    rozdiel očakávanej triedy od triedy, ktorú určil klasifikátor: 1499 určil zle
    úspešnosť klasifikátora = 70 %



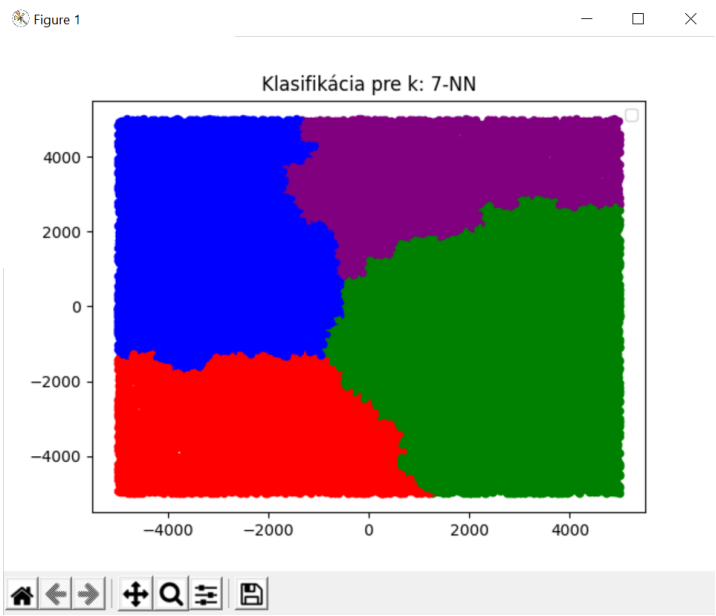
16. počet bodov: 5 000
    k = 15
    čas trvania : 2 minúty
    rozdiel očakávanej triedy od triedy, ktorú určil klasifikátor: 1 646 určil zle
    úspešnosť klasifikátora = 67 %



17. počet bodov: 30 000
    k = 3
    čas trvania : 58 minút
    rozdiel očakávanej triedy od triedy, ktorú určil klasifikátor: 9 629 určil zle
    úspešnosť klasifikátora = 66 %



18. počet bodov: 40 000
    k = 7
    čas trvania : 1 hodina 28 minút
    rozdiel očakávanej triedy od triedy, ktorú určil klasifikátor: 12 685 určil zle
    úspešnosť klasifikátora = 68 %



ZHODNOTENIE TESTOVANIA

Testovala som 2krát pre počet bodov 20 000, raz pre 11 000, raz pre 5 000 a raz aj pre 30 a 40 tisíc. Pre rôzne k (1, 3, 7, 15) bol čas trvania pre 20 000 bodov do pol hodiny. Čím bolo k vyššie, trvalo to o pár minút dlhšie (o minútu, dve). Pri k 15 to niekedy už trvalo ku koncu kratšie, pretože ak mi ostali len 2 farby, vyberanie trvalo kratšie ako keď klasifikátor vyberá zo 4 farieb. Pre 11 000 to trvalo do 10 minút a pri 5 000 bodoch to trvalo do 2 minút. Pri týchto testovaniach boli všetky pomerne úspešné pre k = 1, 3, 7. Niekde sa chyby prejavili trochu viac, pretože keď sa už hneď na začiatku vyskytla nejaká chyba, ďalej sa už iba zväčšovala. Pri k = 15 mi v každom prípade nad 11 000bodov pohltilo minimálne 1 farbu a ostali vo výsledku iba 3 alebo 2 farby. Pre 30 000 bodov som skúsila otestovať k = 3 a pre 40 000 bodov som skúsila otestovať k = 7. Oba prípady vyšli veľmi dobre s úspešnosťou klasifikátora okolo 70%.

SPUSTENIE PROGRAMU A IMPLEMENTÁCIA

Môj program som implementovala v jazyku Python a v programe Pycharm (verzia 2020.2.3) a je spustiteľný. Na spustenie nie sú žiadne špeciálne inštrukcie, jedine treba nastaviť vo for cykle počet bodov (20 000, najviac som skúšala 40 000) s koľkými to chceme spustiť a nad for cyklom počet k. Takisto to isté číslo, ktoré dáme do cyklu treba dopísať aj do počítania úspešnosti klasifikátora.

ZHODNOTENIE A VYLEPŠENIE

Môj klasifikátor po celkovom zhodnotení je podľa mňa úspešný. Očakávaná úspešnosť pravdepodobnosti pre k = 1, 3 a 7 bola okolo 70% a vyššie, pre k =15 to bolo okolo 40 až 60%. Toto môj program spĺňa a niekedy dáva aj lepšie výsledky, závisí to od vygenerovaných bodov. Najlepšie hodnoty mi ukazuje klasifikátor pre k = 1 a k = 3 a čím je k vyššie číslo, tým viac sa klasifikátor zhoršuje. Výhodou K-NN algoritmu je jednoduchosť, no na druhej strane je výrazne pomalý ak ráta s veľkým počtom bodov. Celkový čas trvania by sa dal vylepšiť, keby som implementovala KD stromy. Skrátilo by to čas trvania aj o 3/4. Pre k = 15 by sa to možno dalo vylepšiť tým, keby počiatočných bodov v každej triede nebolo iba 5 ale viac napr. 20 a výsledky klasifikátora by tým pádom boli bližšie k očakávaným výsledkom.