



---

## Zadaci

---

**Zadatak 1.** Učitajte podatke iz prošle zadaće ( $X_a$  i  $y_a$  te  $X_b$  i  $y_b$ ) te nove podatke  $X_c$  i  $y_c$ . Postavite SVM modele koje ćete učiti na učitanim podacima. Koristite ugrađene klase iz modula *sklearn.svm*. Pratite sljedeće upute:

1. Primijenite model SVC s linearnom jezgrenom funkcijom na učitane podatke
2. Ispišite koeficijente  $\theta_0$  i  $\theta$ .
3. Ispišite potporne vektore.
4. Izračunajte širinu dobivene margine.
5. Grafički prikažite podatke, dobivenu hiperravninu koja razdvaja podatke i potporne vektore.
6. Usporedite rezultate iz prošle zadaće s ovom. Odgovorite - u kakvom su odnosu dobivene vrijednosti margina u prošloj zadaći i ovoj?

( Učinite to za podatke  $X_a$ , podatke  $X_b$  i  $X_c$ . )

Odaberite jedan od dva skupa podataka ( $X_a$  ili podatke  $X_b$ ). Što će se dogoditi ako iz njega maknete određeni dio podataka? Koristeći *numpy.random.choice* (ili bilo koju drugu metodu), nasumično odaberite 1000 podataka te na njima ponovite gornji postupak. Usporedite rezultate s gore dobivenim (parametre, margine, potporne vektore). Što možete zaključiti o ulozi potpornih vektora?

Implementirajte funkciju *hinge(x, y)* koja računa broj krivo određenih predikcija modela za skup podataka  $x$ . Testirajte funkciju na vlastitom primjeru (od npr. 4 – 5 podataka). Sada proučite ugrađenu *hinge.loss* funkciju [Link] te ju pokrenite na primjeru kojeg ste odabrali, rezultati vam se moraju poklapati. Također, pokrenite vašu funkciju na nekom od SVC modela koje ste implementirali prethodno u zadatku.

**Zadatak 2.** Učitajte podatke  $X_d$ ,  $y_d$ . Koristeći ugrađeni model SMV-a klasificirajte učitane podatke. Ispišite dobivene koeficijente i izračunajte širinu dobivene margine. Grafički prikažite podatke i dobivenu hiperravninu. Među podacima možete uočiti stršeće vrijednosti, kako one utječu na SVM?

**Zadatak 3.** Učitajte podatke  $X_e$ ,  $y_e$ . Podijelite skup primjera na skup za učenje i skup za testiranje u omjeru 80%:20%. Trenirajte SVM model na podacima za učenje, zatim ispišite dobivene koeficijente i izračunajte širinu dobivene margine. Ispitajte točnost modela na skupu za testiranje koristeći prethodno implementiranu *hinge* funkciju. Postupak ponovite više puta (npr. 10) te uprosječite rezultate. Kao i u prethodnim zadacima grafički prikažite podatke i dobivenu hiperravninu. Možete primijetiti kako ovi podaci nisu linearno separabilni, kako to utječe na SVM?

**Zadatak 4.** Neka je zadana funkcija  $g(z) = \frac{1}{1+e^{-z}}$ . Pokažite kako je  $1 - g(z) = g(-z)$ .



**Zadatak 5.** Izračunajte gradijent *maximum likelihood* kriterijske funkcije koju ste vidjeli na predavanju.

**Zadatak 6.** Neka su zadani podaci  $X \in \mathbb{R}^{m \times n}$  i pripadne izlazne vrijednosti  $y \in \mathbb{R}^{m \times (k-1)}$  koje označavaju pripadnost nekog podatka jednoj od klasa  $0, 1, \dots, k-1$ . Na ovim podacima možemo provesti proces učenja klasifikacijskog modela. Jedan od takvih modela je logistička regresija. Logistička regresija kao model funkciju koristi  $h_\theta(x) = \frac{1}{1+e^{-\theta^T x}}$ , gdje su  $\theta$  parametri koje model uči. Znamo kako se učenje svodi na minimizaciju funkcije  $J(\theta) = \frac{1}{m} \sum_{i=1}^m [-y^{(i)} \log(h_\theta(x^{(i)})) - (1 - y^{(i)}) \log(1 - h_\theta(x^{(i)}))]$  te kako do rješenje tog minimizacijskog problema možemo doći gradijentnom metodom.

1. Implementirajte gradijentnu metodu za slučaj binarne klasifikacije ( $k = 2$ ). Neka vaša funkcija koja pokreće gradijentnu metodu kao argumente prima stopu učenja  $\alpha$  i broj iteracija metode.
2. Isprobajte implementiranu funkciju na podacima  $X_a, y_a$ . Odaberite proizvoljno stopu učenja  $\alpha$  i broj iteracija.
3. Isprobajte implementiranu funkciju na podacima  $X_c, y_c$ . Kako se logistička regresija ponaša prema podacima koji imaju stršeće vrijednosti? Hiperparametre  $\alpha$  i broj iteracija odredite koristeći metodu *pretraživanja rešetke*. Napravite rešetku koja će biti definirana vrijednostima  $\alpha$ , npr. između 0.0001, 0.1 i brojem iteracija između 100, 200. Za svaki par hiperparametara izračunajte točnost modela (accuracy) i odaberite onaj model s najvećom točnošću.
4. Isprobajte implementiranu funkciju na podacima  $X_d, y_d$ . Dodatno nacrtajte graf promjene funkcije cilja kroz iteracije. Opišite dobiveni graf.

Accuracy koristite kao što je navedeno: [https://scikit-learn.org/stable/modules/generated/sklearn.metrics.accuracy\\_score.html](https://scikit-learn.org/stable/modules/generated/sklearn.metrics.accuracy_score.html)

