

1. Atklātās mākslīgā intelekta olimpiādes satura programma

1 Olimpiādes pārskats

Olimpiāde sastāvēs no divām kārtām:

- 1.1 1. kārtā — tiešsaistes kvalifikācija, no 2. līdz 7. februārim.
- 1.2 2. kārtā — klātienē fināls, 3. martā, Rīgā (precīza vieta tiks paziņota vēlāk). Ilgums: 5 stundas.

Reģistrācija un papildu informācija ir pieejama vietnē <https://omi.lv/>.

2 Olimpiādes programmas kopsavilkums

Olimpiāde ir veidota tā, lai būtu pieejama skolēniem, ņemot vērā, ka mākslīgais intelekts un mašīnmācīšanās (ML) nav mācāmi kā atsevišķi priekšmeti skolās. Līdz ar to padziļinātas ML teorētiskās zināšanas netiks prasītas.

Tomēr skolēniem tiek rekomendēts praktizēt dažādus ML uzdevumus pirms olimpiādes, jo praktiska pieredze būtiski uzlabos rezultātus.

Tāpat skolēniem ir ļoti ieteicams **justies brīvi programmējot Python un izmantojot NumPy, Pandas un Scikit-learn bibliotēkas**, kā arī zināt **pamata pārskatu par matemātisko teoriju ML un zemāk norādītajiem ML algoritmiem.**

3 Olimpiādes programma

Olimpiāde aptvers šādas tēmas:

- 3.1 Mašīnmācīšanās pamati
- 3.2 Klasiskie mašīnmācīšanās algoritmi
- 3.3 Neironu tīkli un dziļā mācīšanās
- 3.4 Mašīnmācīšanās pielietojumi

Tēmas ar prefiksu **[A]** ir **stipri rekomendētas kā pamatzināšanas** olimpiādes sagatavošanai, savukārt tēmas ar prefiksu **[B]** nav **obligātas**, taču tās var **ievērojami uzlabot problēmrisināšanas spējas.**

Dažas tēmas tika atstātas angļu valodā, jo šī terminoloģija starptautiski tiek lietota tieši angļu valodā un tās tulkošana varētu radīt neprecizitātes vai apgrūtināt mācīšanos.

4 Foundations of Machine Learning

4.1 Programmēšanas pamati

- 4.1.1. [A] Python programmēšanas pamati
- 4.1.2. [A] NumPy un Pandas bibliotēkas
- 4.1.3. [A] Scikit-learn ML bibliotēka
- 4.1.4. [A] Spēja izmantot sniegtās ML bibliotēkas

4.2 Matemātiskie pamati

- 4.2.1. [A] Polinomu, pakāpju un logaritmu izpratne
- 4.2.2. [A] Spēja analizēt dotu funkciju pamatuzvedību
- 4.2.3. [A] Augsta līmeņa pārskats par Bajesa varbūtību teoriju
- 4.2.4. [B] Lineārā un loģistiskā regresija

4.3 ML teorija

- 4.3.1. [A] Regresija un klasifikācija
- 4.3.2. [A] Nepietrenēts un pārtrenēts modelis (Underfitting, Overfitting)
- 4.3.3. [A] Krustotā validācija (Cross-Validation)
- 4.3.4. [B] Jaukšanas matrica un ROC līknes (Confusion Matrix and ROC Curves)

5 Klasiskie mašīnmācīšanās algoritmi

- 5.1 [A] Vispārīga izpratne par tālāk minētajiem algoritmiem
- 5.2 [B] Lineārā un loģistiskā regresija
- 5.3 [B] K-Nearest Neighbors (KNN)
- 5.4 [B] Support Vector Machines (SVM)
- 5.5 [B] Decision Trees
- 5.6 [B] Random Forest
- 5.7 [B] Naīve Bayes
- 5.8 [B] K-means Clustering
- 5.9 [B] Principal Component Analysis (PCA)

6 Neironu tīkli un dziļā mācīšanās

- 6.1 [A] Vispārīga izpratne par tālāk minētajiem algoritmiem
- 6.2 [B] Perceptrons
- 6.3 [B] Multi-Layer Perceptrons (MLP)
- 6.4 [B] Gradient Descent & Backpropagation
- 6.5 [B] Aktivācijas funkcijas
- 6.6 [B] Pastiprināšanas mācīšanās pamati (Reinforcement Learning basics)
- 6.7 [B] Computer Vision

7 Mašīnmācīšanās pielietojumi

- 7.1 [A] Datu sagatavošana
- 7.2 [A] Modeļu novērtēšanas metrikas
- 7.3 [A] Hiperparametru regulēšana (Hyperparameter Tuning)
- 7.4 [B] Modeļu ansambļi (Model Ensembles)
- 7.5 [B] Feature Engineering
- 7.6 [B] Data Imputations
- 7.7 [B] Transfer Learning
- 7.8 [B] Datu apstrāde
- 7.9 [B] ML ētika

Sveiks, lasītāj! Ja esi izlasījis līdz šai vietai — apsveicam!

Vēlamies atstāt nelielu mājienu gatavošanās procesam olimpiādei.

Klejo baumas, ka viens no matemātikas uzdevumiem varētu būt saistīts ar ļoti vienkārša MLP aktivācijas funkcijas aprēķināšanu: divi slāņi, viens neirons katrā slānī, abos izmantota viena un tā pati aktivācijas funkcija. Lai gan precīzas detaļas nav zināmas, viens noderīgs novērojums ir tas, ka šāda MLP aprakstītā kopējā funkcija ir lineāra.

Tātad jautājums ir šāds: kā tu pieietu sākotnējās aktivācijas funkcijas noteikšanai?