

3-Dars's report

1.ML model yaratish

2.Kaggle/Github

3.Data bilan ishlash

4.Conditions

AI o'rganishni boshladik uni ichidan ML ni tanladik va uni ichki qismlarini ko'rib chiqdik . Ichki qismlari Supervised, unsupervised va reinforcement Learnings larga bo'linadi va ularni aloxida vazifasi bor. Biz hozir supervised bo'limi o'rganayapmiz va u 2 ga bo'linadi . Biz uni farqlash uchun avval datasetimizga qarashimiz kerak va qiymatlarga etibor beramiz ularda har xil classlar bor misol uchun umumiy classlar soni 2ta va NO/YES va endi bizni Supervised learningimizni 1-bo'limi Classification deyiladi . Agar cheksiz bo'lsa bu regression deyiladi bu qism Supervised learningni 2-qismi. Endi biz yaxshilab SL ga qarash u 2qisimga bo'linadi. Classificationni olsak qiymat sanoqli bo'lishi kerak, regression ni olsak qiymat continue yani davomli bo'lishi kerak, biz ularni farqlashimiz uchun javobga yani class lardagi Target qiymatlarga qaraymiz. Endi Classificationni o'zi ham 2 ga bo'linadi 1-Binary va 2-multi-class . Binary faqat javob 2ta bo'lishi kerak , Multi-classda esa 3 yoki undan ko'p bo'lishi kerak. For example; Foodballcilar o'yin davomida jarohat oladimi yoki yoq bu ML ni SL gini Classification qismidagi Binary siga to'g'ri keladi , lekin qaysidir jarohat turi haqida bo'lsa misol uchun; Tizza j, qo'l j, va h.k lar bo'lsa bu Multi-class qismiga kiradi. Biz bu class lar sonini topish uchun codingdan foydalanamiz . biz ketka ket va tartib bilan qilishimiz kerak . Albatta bizga Dataset kerak bo'ladi biz datasetni qayerdan olamiz degan savol bor biz , Dataset larni Github/Kaggle lardan olamiz va datasetimiz ocan ipynb faylimiz bilan bir joyda yani yonma yon turishi kerak. Birinci o'rinda biz Pandasni isntall qilishimiz kerak yani kutubxona o'rnatishimiz kerak . bu Cod [import pandas as pd] dan foydalanamiz bu cod da [As] ni vazifasi Pandas ni pd qilib ishlatishimiz uchun yani qulaylik uchun yordam beradi . Bizga 1-o'rinda kutubxonalar Install qilishimiz kerak, bizga shu codlar yordam beradi.

1. # pip install pandas

2. # %pip install pandas

3. # !pip install pandas

Biz agar koproq coding qilmoqchi bo'lsak oddiy oynada bu cod sal xalaqit qiladi biz Camfort uchun Terminal qismiga o'tib shu cod dan foydalanamiz . Eni biz datani tortib olishimiz uchun yani codinga tortishimiz uchun coddan foydalanamiz [df=pd.read_csv('datani copy relative path qilib shuyerga joylashtiramiz ')] va dataset o'rniadi. Enid biz bemmalol cods lardan use qilishimiz mumkin Such as: 1-Cod va eng kop ishlatiladigan codimiz [# df.head()] biz bu cod orqali datani boshidagi 5 ta yoki 3 yoki h.k qatorgaca so'rasak bizga ciqarib beradi. Agar biz tagidan kelmoqchi bo'lsak 2-Cod [# df.tail()] dan foydalanamiz bizga qaysi qatorgaca kelsa () ishiga kerakli bo'lgan qatorgaca so'raymiz. Agar bizga har bir class dagi ustunlar soni kerak bo'lsa biz [# df.nunique()] kodidan foydalanamiz bu kod 3-Codimiz. Nunique qiymat duplicate qiymat. Agar bizga faqat bitta ustun kerak bo'lsa biz [# df.['ustun nomi']] dan yani 4-Codimizdan foydalanamiz. Agar bizga 2/3 va undan ko'p ustun kerak bo'lsa biz [# df[['ustun nomi','ustun nomi']]] bu bizni 5-Codimiz.

Kaggle/Github ga keladigan bo'lsak biz u saytlar orqali xar xir Datasets topishimiz mumkin va har hil hajmda bizga qulay bo'lishi uchun ko'proq csv fayl ko'rinishida bo'lsa yaxshi bo'ladi. Yana bir narsaga etibor berishimiz kerak dataset bilan ochgan faylimiz yonma yon bo'lishi kerak biz ko'roq datasetni sifililigiga eribor qaratishimiz kerak agar kompyuter tushunmaydigan tilda bo'lsa biz tushunadigan tilga o'girib olishimiz kerak yani Canvert qilishimiz kerak yoki yanngi coding qilishimiz kerak

Data bilan ishlashda bizga kerakli kodlar bor misol uchun: # df.head() dan kora # da.info() ishlatib bizga yanada cuqurroq ma'lumot beradi yani umumiy ma'lumot beradi. Biz yana u cod lar orqali data types necta ekanligini va nimalar qantashganini yani umumiy malumot beradi shu cod. Yana bizga berilgan ma'lumotlarda biz ajrata ola olishimiz kerak misol ucun: biz Pandas kutubxonasida ishlayotganimiz yani shu orqali data olganimiz , keyingisi biz yana ahamiyat beradigan narsa entries soni (531) , yana Data columns yani data ustunlari ga ham etibor beramiz misol: 20 ta ustun bor bu degani 19 ta input va 1 ta output . Bizni ustunlarimiz 0 dan boshlanadi. Yana bir ma'lumot Non-null cout --> bu tushub qolgan qiymatlar bormi yoki yoq shuni ko'rsak boladi. Agar hammasi 531 bo'lsa tushib qolgan qiymat bo'lmaydi. Endi biz Dtype ga e'tibor bersak ularni ichida Data types lar bor 1.Int 2.Float 3.Str 3.Boolean larni o'z ichiga oladi. Bular ustundagi qanca malumotlarni olganligiga qarab har xil bo'ladi misol: int64/32/16/8 va deyarli boshqa types lar ham shunday , str=object ga , biz tushub qolgan qiymatlarni ko'rishimiz uchun [df.install().sum()] shu coddan foydalanamiz.

Yana bir cod bor [`df.install()`] lekin bu kod anca qiyin va noqulay yani bizga T/F ko'rinishida ciqaradi, uni o'rniga [`df.install().any()`] qilsak bizga anca comfort qiladi yani tushib qolgan qiymat fulse yoki true db ciqaradi yani boolean orqali topish deyiladi

Conditions ---> ma'lumotlarni (Data) boshqarish, nazorot qilish va filterlash uchun ishlatiladigan codlar bloki bu Conditions IF -> THEN misol ucun: Agar son + bo'lsa yaxshi aks holda yomon degan so'zni ekranga ciqaring biz boshqa kodlar orqali yozamiz : `if`, `elif`, `else` lardan tashkil topgan.