8-dars: Model Building

1. **Missing values bilan qanday ishlaymiz? :** Avval qaysi ustudna nechtadan missing values borligini aniqlaymiz: **df.isnull().sum()** yoki **df.isna().sum()**, keyin qaror qabul qilamiz qaysi usullardan foydalanishni, masalan mean, mode, median, fixed yoki 2 xil drop bilan ishlashni.

1.1 **Mean** nima va kodini aytib bering? : **Mean** bu o'rta arifmetik qiymat bilan missing bo'lgan qiymatlarni to'ldirish (faqat **numerical**), bunda o'sha ustundagi barcha qiymarlarini o'rta arifmetiki olinadi. Kodi esa: **df[col].fillna(df[col].mean(), inplace=True)** orqali to'ldiriladi.

1.2 **Mode** nima va kodini aytib bering? : **Mode** bu ustundagi eng ko'p takrororlangan qiymat bilan to'ldirishdir (**categorical** ham **numerical** uchun ham ishlatiladi). Kodi esa **df[col].fillna(df[col].mode()[0], inplace=True)**.

1.3 Shu yerdagi **mode()[0] -[0]** nega qo'yildi va nima vazifani bajaradi? : Buning asosiy vazifasi shundaki mode-eng ko'p takrorlangan qiymatlarni seriesda ko'rinishda qaytaradi [0] esa eng birinchi ya'ni eng ko'p taktorlanaganlarning iichidagi eng birinchisin oliib to'ldir degan bo'lamiz.

1.4 **Median** nima va kodini aytibi bering?: **Median** bu ustinning eng o'rtasidagi qiymati bilan to'ldirishdir (faqat **numerical** uchun ishlatiladi). Kodi: **df[col].fillna(df[col].median(), inplace=True)**.

1.5 Dropning 2 xil usuli bor ular qanday va nima uchun kerak? : 1-si bu **qator** bo'ylab tashlash masalan bizda ma'lum bir qiymatlar tushib qolganda butun qator bo'ylab drop qilamiz, bunda datasetimiz kattaligi, o'sha qator o'ta muhim bo'lmaganda va missing qiymatlar kam bo'lganda qo'l keladi. 2-si esa **ustun** bo'ylab tashlab yuborish bunda ustunimizda juda ko'p qiymatlar missing bo'lganda, suniylikdan qochish uchun, va real natijani yana ham suniylashtirmaslik uchun undan butunlar qutulish maqbulroq deb ko'riladi. Kodlari: Row drop: **df.dropna(inplace=True**), Col drop**: df.drop(columns=['column name'], inplace=True).**

2. One-hot va **Label encoding** orasida eng asosiy farq nima?: One-hot encoding bu categorical qiymatlarni mashina modellari tushunadigan **ikkilik (0 va 1)** ko'rinishiga o'tkazib beradi. **Label encoding** esa bu **categorical** qiymatlarni **alifbo** tartibidagi ketma-ketlikda **0** dan to **z** gacha nechta qiymat bo'lsa o'shatgacha almashtirib beradi.

2.1 **One-hot encoding** qilganimizda nima bo'ladi?: u ustundagi **number of uniquedan** kelib chiqib har bir classni alohida ustunga aylantiradi. Asosan biz **df.nunique()** orqali tekshirib undan so'ng one-hot yoki label qilishlikka qaror qilamiz.

2.2 One-hot encodignda **threshold** tushunchasi bor shu niam vazifani bajaradi? : **Threshold** bu limit degani ya'ni agar limit qo'ymasak kelgusida

ustunlar soni qatorlar sonidan ko'payib **collapse** holiga kelib qolishi oldini olish uchun kerak bo'ladi.

2.3 **Encodingni** qanday chaqiramiz va **one-hot encoding** kodi qanday?: **from sklearn.preprocessing import LabelEncoder, encoder = LabelEncoder(), if df[col].dtype == 'object':**
va yana **if df[col].nunique() <= 5:** deya **threshold** va undan keyin quyidagilarni yozamiz: **dummies = pd.get_dummies(df[col], prefix=col, dtype=int) and df = pd.concat([df.drop(columns=[col]), dummies], axis=1).**

2.4 Label encoding kodi qanday yozilar edi?: **df[col] = encoder.fit_transform(df[col])** deya.

2.5 Shu yerdagi fit_transform() nima fazifani bajaradi?: **fit**--bu katregoriyalarni **mapping** ko'rinishida o'rganadi, **transform** esa shu o'rganilgan **mapping** ma'lumotni son ko'rinishiga o'tkazadi.

3. Scaling nima? : Scaling bu qiymatlar orasidagi farq juda katta va juda kichik bo'lganda qiymatlarni birlashitirish uchunu ishlatilinadigan Data Preprocessing bo'limidir.

3.1 Ustunlar object bo'lsa uni scaling qila olamizmi va qaysi oraliqda scalingqilish shart emas?: Yo'q ustunlarimiz faqat numerical bo'lishi kerak va **[0, 1]** oraliqda bo'lsa scaling qilish shart emas.

3.2 **Scaling** qanday chaqiriladi va uning qanday turlarini hozirgacha bilasiz?: **from sklearn.preprocessing import MinMaxScaler**, **StandardScaler, RobustScaler, scaler = MinMaxScaler() or scaler = StandardScaler() or scaler = RobustScaler()** holtda chaqiriladi va turlari bor.

3.3 **MinMaxScaler** nima?: qiymatlari **[0, 1]** oralg'ida **scaling** qiladi va musbat ko'rinishida bo'ladi.

3.4 **StandardScaler va RobustScaler** ularning qanday o'xshash farqi bor? : har ikkisda scaling qilinganda qiymatlar **manfiy** ko'rinishida bo'lishi mumkin.

3.5 **Scaling** kodini aytib bering? : **df[col] = scaler.fit_transform(df[col])** deya chaqiriladi, ammo undan oldin esa biz **num_cols = df.select_dtypes(include=['data turlarini har birini bittadan yozib chiqish kerak']).columns.drop('taget qiymat')** qilishimiz kerak sababi bizda target qiymat scaling bo'lmaydi.

4. Preprocessing nima?: Preprocessing bu **raw** (xom) datani model tushunadigan holga olib kelish jarayoni.

4.1 **Missing** qachon qilinadi? : **datasetni df.info()** qilib va **df.insull().sum()** orqali tushib qolgan qiymatlari mavjuda bo'lsa, agar to'liq bo'lsa bu jarayon taslab ketiladi.

4.2 **Encoding** qachon qilinadi va nimaga e'tibor beriladi?: **Missing** qiymatlarni **handle** qilingandan so'ng, ularni tekshirib **df.isnull().sum()** va **df.nunique()** orqali va undan keyin qaror qabul qilinadi **one-hot** yoki **label encoding** deya.

4.3 **Scaling** qachon qilinadi?: **Encoding** qilindan so'ng, **df.head()** orqali tekshirilib, baho berilaib maqul **scaling classlaridan** foydalangan holda.

4.4 **Model** qachon **train** va **test** qilinadi? : **3 la bosqich** muvaffiqatli o'tkandan so'ng, **from sklearn.model_selection import train_test_split,** undan oldin esa **x = df.drop('target qiymat', axis=1), y = df['target qiymat'],** unda keyin esa **x_train, x_test, y_train, y_test = train_test_split(x, y, test_size=0.2 or 0.3 datasetni kattaligi yoki kichikligiga qarab, random_state=42).**

4.5 Bu yerdagi r**andom_state=42** nima va nima vazifa bajaradi?: bu **modelni stabil** ishlashini taminlaydi. **Stabil ishlash nima deganda esa**: kodga hech qanday o'zgarish kiritilmaganda ham bir hil natija berishi modelni stabilligi bo'ladi.