Përshkrimi i Punës

Pikat Kryesore

1. Forma e re e datasetit (74706, 78)
2. Trajtimi i vlerave që mungojnë (Nuk mungon më asnjë vlerë).
3. Shtimi i kolonave ku ka pasur nevojë.
4. Shndërrimi i vlerave të kolonave në kategorike, numerike etj., sipas nevojës.

Rrjedha e Punës

Së pari kemi bërë analizën e datasetit që e kemi marrë, pra 'survey_results_public.csv'. Dataseti ka përmbajtur 89184 rreshta dhe 84 kolona, ku shumë prej tyre kanë pasur vlera që mungojnë (null values).

Kemi parë që dataset-i nuk ka pasur vlera të përsëritura (duplikate) dhe, së pari, kemi fshirë kolonat dhe rreshtat tek të cilat kanë munguar më shumë se 50% e informatave. Pra, një rresht ku 50% e kolonave kanë qenë të zbrazëta është fshirë. Ose ngjashëm, një kolonë në të cilën nuk ka pasur të dhëna për 50% të rreshtave është fshirë. Kodi për rreshta:

```
threshold = 0.5

missing_ratio = clean_df.isnull().mean(axis=1)

rows_to_drop = clean_df[missing_ratio > threshold].index

clean_df = clean_df.drop(rows_to_drop)
```

Kështu kemi fituar formën e re të datasetit: (74706, 71)

Kolona Q120 është identifikuar si kolonë e panevojshme (redundante). Ka përmbajtur vetëm një vlerë 'I agree' dhe si pasojë nuk është e dobishme për të nxjerrë informata nga pjesëmarrësit e datasetit. Vetëm prania e pjesëmarrësit në dataset nënkupton se ai/ajo janë pajtuar të jenë pjesë e kësaj 'survey'. Kështu kjo kolonë është fshirë.

Te kolona MainBranch kemi përmirësuar lexueshmërinë e saj dhe e kemi shndërruar në një kolonë me vlera kategorike. Ngjashëm kemi vepruar edhe për Age dhe EdLevel.

```
clean_df['MainBranch'] = clean_df['MainBranch'].replace({
    'I am a developer by profession':'professional developer',
    'I am not primarily a developer, but I write code sometimes as part of my work/studies': 'not primarily a developer',
    'I am learning to code':'aspiring developer',
    'I code primarily as a hobby':'codes for fun',
    'I used to be a developer by profession, but no longer am':'former-developer'
})
```

```
clean_df['MainBranch'] = clean_df['MainBranch'].astype('category')
```

Për kolonën Employment kemi vepruar si në vijim:
Kolona ka pasur vlera që mungojnë. Këto janë të pakta, andaj i zëvendësojmë me moden që vlen për kolonën e caktuar. Kjo kolonë përmban vlera string të cilat janë të ndara me ‘;’ dhe paraqesin pozita të ndryshme të cilat i mbajnë pjesëmarrësit e kësaj survey.

Këtë e trajtojmë duke e shndërruar së pari në një listë, ku elementët ndahen nga njëri-tjetri me ‘;’.
Pasi i kemi në një listë, eksplodojmë kolonën në disa kolona të reja që përfaqësojnë vlerat që përmbahen brenda listës së Employment. Më pas këto kolona, varësisht nga paraqitja për një pjesëmarrës të caktuar, marrin vlera 0 dhe 1, të cilat janë më të përshtatshme se listat ose stringjet fillestare. Gjithsej kemi 8 kolona të reja individuale që përfshijnë ‘Employed-Full-Time’, ‘Employed-Part-Time’, ‘Student-Full-Time’ etj., ku një pjesëmarrës mund të ketë vlera 1 në disa nga këto kolona.
Pasi kemi vlera individuale që përfaqësojnë më mirë rolet e pjesëmarrësve, largohet kolona origjinale Employment.

```
employment_mode = clean_df['Employment'].mode()[0]
clean_df['Employment'] = clean_df['Employment'].fillna(employment_mode)
```

```
clean_df['Employment'] = clean_df['Employment'].apply(
    lambda x: [s.strip() for s in x.split(';')] if x else []
)
```

```
exploded = clean_df['Employment'].explode()

one_hot = pd.crosstab(exploded.index, exploded)

clean_df = clean_df.join(one_hot)
```

```
clean_df = clean_df.drop(columns='Employment')
```

Ngjashëm kemi vepruar edhe për kolonën RemoteWork, ku vlerat që mungonin janë zëvendësuar me moden e kolonës.

Kolonat YearsCode dhe YearsCodePro janë shndërruar në vlera numerike.
Për këtë, kemi zëvendësuar vlerat jo-numerike 'Less than 1 year' me '1', dhe 'More than 50 years' me '50', për një interpretim më të thjeshtë.

Sikurse me Employment, disa kolona tjera kanë pasur vlera që ishin lista të koduara si stringje të ndara me ';'. Këto i shndërrojmë në lista të mirëfillta brenda një for loop.

```
cols = [
    'CodingActivities','LearnCode','LearnCodeOnline',
    'BuyNewTool','LanguageHaveWorkedWith','LanguageWantToWorkWith',
    'DatabaseHaveWorkedWith','DatabaseWantToWorkWith','PlatformHaveWorkedWith',
    'PlatformWantToWorkWith','WebframeHaveWorkedWith','WebframeWantToWorkWith',
    'MiscTechHaveWorkedWith','MiscTechWantToWorkWith','ToolsTechHaveWorkedWith',
    'ToolsTechWantToWorkWith','NEWCollabToolsHaveWorkedWith','NEWCollabToolsWantToWorkWith',
    'OpSysPersonal use','OpSysProfessional use','OfficeStackAsyncHaveWorkedWith',
    'OfficeStackAsyncWantToWorkWith','OfficeStackSyncHaveWorkedWith','OfficeStackSyncWantToWorkWith',
    'AISearchHaveWorkedWith','AISearchWantToWorkWith','NEWSOSites', 'ProfessionalTech'
]

for col in cols:
    clean_df[col] = df[col].fillna('').apply(
        lambda x: [item.strip() for item in x.split(';') if item.strip()]
    )
```

Shumë nga vlerat që mungonin në dataset janë zëvendësuar:

- me medianën, nëse ishin numerike,
- me moden, nëse ishin kategorike,
- ose me metodën ffill në raste të rralla, ku kemi plotësuar vlerat me atë që vjen më pas.

Shembuj të trajtimeve të veçanta:

Currency:
Te kolona Currency kishim vlera që mungonin, por te Country jo. Kjo na ndihmoi sepse ato janë të lidhura ngushtë.
Krijuam një dictionary country_to_currency, ku çelësi është shteti dhe vlera është kodi i valutës.
Vlerat në Currency i plotësuam bazuar në vlerën e Country të rreshtit përkatës, i cili pastaj është bërë map në vlerën e tij përkatëse në dictionary country_to_currency.
Vlerat ishin në formatin 'USD\t United States Dollar' në dataset, mirëpo ato të cilat i kemi zëvendësuar kishin vetëm kodin e valutës së caktuar. Kështu përmes një regex që identifikon

hapësirat tab kemi hequr pjesën që tregon emrin e valutës duke mbajtur vetëm kodin e saj për të gjitha.

```
clean_df['Currency'] = clean_df['Currency'].fillna(
    clean_df['Country'].map(country_to_currency)
)
```

```
import re

clean_df['Currency'] = clean_df['Currency'].apply(lambda text: re.sub(r'\t.*', '', text))
```

CompTotal:
Kishim shumë vlera që mungonin. I plotësuam me medianën e pjesëmarrësve të grupuar sipas shtetit të tyre, sepse rrogat dhe valutat ndryshojnë sipas shtetit. Përdorim medianën sepse është më e qëndrueshme ndaj vlerave abnormale (outliers).

```
global_median = clean_df['CompTotal'].median()

clean_df['CompTotal'] = clean_df.groupby('Country')['CompTotal'].transform(
    lambda x: x.fillna(x.median() if not x.dropna().empty else global_median)
)
```

Kolonat Object → kategorike:
Disa kolona me tipe ‘object’, por që përmbajnë vlera kategorike, i konvertuam në tipe kategorike.

ICorPM:
Mbushim vlerat që mungojnë në bazë të viteve të programimit profesional.
Në bazë të supozimit se personat me më shumë vite përvoje janë menaxherë, të tjerët janë kontribues individualë.

```
threshold = clean_df['YearsCodePro'].mean()

def impute_ICorPM(row):
    if(pd.isna(row['ICorPM'])):
        if row['YearsCodePro'] > threshold:
            return 'People manager'
        else:
            return 'Individual contributor'
    else:
        return row['ICorPM']

clean_df['ICorPM'] = clean_df.apply(impute_ICorPM, axis=1)
```

WorkExp:
Vlerat që mungonin i plotësuam me medianën e personave me të njëjtin numër vitesh programimi profesional.

```
clean_df['WorkExp'] = clean_df.groupby('YearsCodePro')['WorkExp'].transform(
    lambda x: x.fillna(x.median())
)
```

ConvertedCompYearly:
Vepruam njësoj si te CompTotal, me medianën e personave të grupuar sipas shtetit të tyre.

```
global_ccy_median = clean_df['ConvertedCompYearly'].median()

clean_df['ConvertedCompYearly'] = clean_df.groupby('Country')['ConvertedCompYearly'].transform(
    lambda x: x.fillna(x.median() if not x.dropna().empty else global_ccy_median)
)
```

Në fund, kemi fituar një dataset të plotë, pa vlera që mungojnë, me informata të organizuara mirë dhe të strukturuara në mënyrë që të mund të analizohen dhe përdoren në mënyrë efikase.

Ky dokument shërben vetëm si përmbledhje e punës, kodi shërben si përfaqësim më i mirë i punës së bërë.

Punuar nga: Redon Brovina