Task: Preprocesarea datelor magazinului online

Curs: Data Analysis and Processing using Python
Modul: Pregătirea datelor pentru o analiză de succes

Scopul sarcinii:

Dezvoltați un script Python care pregătește datele pentru analiză, incluzând pașii cheie de curățare, transformare și crearea de informații suplimentare care pot îmbunătăți analiza datelor.

Contextul sarcinii:

După orientarea inițială în date, Alex vrea să le pregătească pentru o analiză concretă în vederea îmbunătățirii strategiilor de vânzări. Sarcina voastră este să îl ajutați în etapele de preprocesare: curățare, transformare, procesare a valorilor lipsă, crearea de noi coloane și efectuarea de analize simple.

Sarcina:

Mai întâi, trebuie să descărcați setul de date pe care l-a primit Alex. Este același set de date folosit în prima sarcină:

online_store_data.csv

Sarcina voastră este să scrieți un script Python care să efectueze transformarea datelor de intrare. Pașii care trebuie implementați sunt:

1. Transformarea datelor

1. **convertiți** tipurile de coloane `quantity_sold` și `num_of_ratings` în valori întregi;
2. **convertiți** tipul de coloană `quantity_in_stock` în valoare întreagă;
3. **convertiți** tipul de coloană `date_added` în `datetime`;
4. **extrageți** valorile numerice ale evaluărilor produselor din coloana `rating`;
5. **eliminați** rândurile care nu au valori din coloana `product_name`;
6. **eliminați** rândurile care au valori lipsă în mai mult de 4 coloane;
7. **eliminați** rândurile care sunt duplicate complete, păstrând doar primul rând și eliminând toate duplicatele ulterioare.

2. Ingineria caracteristicilor

- **generați** o caracteristică nouă care ar trebui să arate venitul realizat pe produs (revenue); venitul se calculează atunci când prețul produsului este înmulțit cu numărul total de exemplare vândute.

3. Analiza

După ce ați creat scriptul pentru transformare, trebuie să îl folosiți pentru a realiza transformarea datelor magazinului online. După transformare, trebuie să faceți două analize:

- **găsiți** cele 10 produse din categoria *Keywords* care au generat cel mai mare venit,
- **găsiți** cele 10 produse din categoria *TVs* care au generat cel mai mic venit.

Ghid pentru rezolvarea sarcinii:

Mai jos sunt pașii care vă pot ajuta în realizarea sarcinii:

1. Conversia tipurilor de date

- Pentru a converti tipurile de date, va trebui să folosiți abilități dobândite din lecțiile „Tipuri de date: Fundamentul fiecărei analize

de succes” și „Conversia tipurilor: Transformarea datelor în formatul corect”. Încercați mai întâi abordările de bază (utilizând metoda a `stype()`). Dacă conversia nu poate fi realizată astfel, treceți la abordarea care permite manipularea valorilor nevalide (`to_numeric()`). Nu uitați că puteți utiliza tipuri de date Pandas care permit apariția valorilor `NA` (`Int32`, `Int16`). Pentru a converti o coloană la tipul `datetime`, folosiți metoda `to_datetime()`.

2. Extragerea valorilor numerice din coloana rating

- Pentru a efectua extragerea valorilor numerice din coloana `rating`, mai întâi va trebui să analizați valorile existente în acea coloană. Listați câteva valori aleatorii și încercați să determinați tiparul în care sunt formate. Scopul este să găsiți un mod prin care să extrageți datele numerice din aceste valori. Afișați valorile unice (`unique()`) și încercați să găsiți valori neobișnuite care nu se încadrează în tipar. După această analiză, scrieți o funcție pentru extragerea valorilor numerice. Aplicați funcția pentru fiecare rând folosind metoda `apply()` și scrieți noile valori peste cele vechi (în coloana existentă). Lecția „Date ascunse în text: Parsarea și extragerea datelor numerice” vă poate ajuta în acest proces.

3. Eliminarea rândurilor cu valori lipsă

- Pentru eliminarea rândurilor cu valori lipsă, va trebui să folosiți metoda `dropna()`. Aceasta este o metodă care se aplică asupra obiectului `DataFrame` și, în funcție de parametrii transmiși, poate îndepărta diferite rânduri cu valori lipsă. Pentru a defini coloana în care va fi verificată prezența valorilor lipsă, folosiți parametrul subset. Parametrul `thresh` este utilizat pentru a defini numărul de coloane care trebuie să aibă valori nenule pentru ca rândul să nu fie eliminat.

4. Eliminarea duplicatelor

Pentru a elimina duplicatele, folosiți metoda `drop_duplicates()`. Parametrul `keep` vă poate ajuta să definiți care rânduri vor fi păstrate în cadrul setului de date (primul, ultimul etc.).

5. Ingineria caracteristicilor

Pentru a genera o nouă coloană cu venitul total obținut de fiecare produs, va trebui să folosiți abordările din lecția „Generarea de noi date: Ingineria caracteristicilor în acțiune”. Noua coloană trebuie să aibă numele revenue, iar valorile sale vor fi obținute prin înmulțirea valorilor din coloanele `quantity_sold` și `price`. Pentru a realiza acest lucru, este suficient să scrieți o singură linie de cod.

6. Găsirea celor 10 tastaturi cu cel mai mare venit și a celor 10 televizoare cu cel mai mic venit

După ce ați realizat transformarea datelor, trebuie să efectuați două analize. Condiția prealabilă pentru ambele analize este că ați creat anterior o nouă coloană care arată venitul obținut de fiecare produs (`revenue`). Valorile din această coloană sunt esențiale pentru efectuarea analizelor solicitate. Cele 10 tastaturi cu cel mai mare venit le veți găsi prin filtrarea produselor astfel încât să rămână doar cele din categoria *Keyboards*. Apoi, veți sorta produsele obținute după coloana revenue în ordine descrescătoare și cele mai mari venituri vor fi în partea de sus. Pentru a găsi cele 10 televizoare cu cel mai mic venit, trebuie să urmați aceiași pași cu două modificări. Filtrarea va fi făcută astfel încât să rămână doar produsele din categoria *TVs*, iar sortarea va fi efectuată în ordine crescătoare. Astfel, produsele cu venitul cel mai mic vor apărea la începutul setului de date.

Predarea sarcinii:

După finalizarea sarcinii, codul sursă al proiectului complet trebuie transmis instructorului cursului.

- Denumirea fișierului: `task2_data_preprocessing.py`
- Se recomandă structurarea codului în funcții și să conțină comentarii.
- Trimiteți o arhivă `.zip` sau `.rar` cu un fișier `.py` prin intermediul platformei.

Cerințe pentru evaluare

Instructorul va evalua sarcina după următoarele criterii:

Criteriu	Procent din notă
Conversie corectă a tipului de date	20%
Extragerea unei valori numerice din coloana rating	20%
Eliminarea rândurilor cu valori lipsă	15%
Eliminarea duplicatelor	10%
Crearea unei coloane noi revenue	15%
Calitate și lizibilitatea codului (comentarii, structura)	20%

Mod de evaluare

□□□□□ **(90-100%)**: Sarcina este completă, toate calculele și intrările de date sunt corecte, rezultatele sunt afișate clar, iar codul este lizibil și scris eficient.

□□□□ **(75-89%)**: Erori minore la introducerea datelor sau la calcule, dar majoritatea funcționalității funcționează corect. Prezentarea rezultatelor este clară, cu câteva erori minore.

□□□ **(60-74%)**: Mai multe erori în introducerea datelor, calcule sau afișarea rezultatelor. Codul conține erori minore de logică sau de sintaxă.

□□ **(50-59%)**: Erori majore în introducerea datelor și calcule. Programul funcționează doar parțial, iar afișarea rezultatelor nu este corectă.

□**(45-49%)**: Sarcina este parțial finalizată, dar există erori semnificative în introducerea datelor și în calcule, care afectează acuratețea rezultatelor. Programul are o funcționalitate limitată și necesită îmbunătățiri pentru a

afișa corect informațiile solicitate.

 (0-44%): Programul nu este funcțional, conține erori grave care interferează cu execuția sa sau sarcina nu a fost înțeleasă corect.

Succes cu sarcina!