Nama : Putri Amelia Suhendi

Nim : 1103213195

Tugas Ml

**Penjelasan Matematis: Analisis Missing Values**

**1. Identifikasi Data Kosong**

Langkah awal dalam menganalisis data yang hilang adalah mengenali sel-sel kosong dalam setiap kolom.  
Hal ini dilakukan dengan metode logika biner, di mana:

* Nilai kosong dianggap sebagai **True** (atau 1)
* Nilai terisi dianggap sebagai **False** (atau 0)

Setelah itu, dilakukan penjumlahan nilai-nilai True untuk menghitung berapa banyak entri yang hilang di setiap kolom.  
Secara matematis, ini serupa dengan menggunakan fungsi indikator:

White text on a black background

AI-generated content may be incorrect.

**2. Menghitung Proporsi Kehilangan Data**

Agar lebih mudah dipahami, jumlah data yang hilang biasanya dikonversi ke bentuk persentase dari total baris data.  
Rumus dasarnya adalah:

A black and white text

AI-generated content may be incorrect.

**Contoh Kasus**:  
Jika kolom director mengandung 10 entri kosong dari total 100 baris, maka:

A black background with white text

AI-generated content may be incorrect.

**3. Mengurutkan Berdasarkan Tingkat Kehilangan**

Setelah semua nilai dihitung, kita dapat mengurutkannya berdasarkan tingkat kehilangan data.  
Dengan menyusun kolom dari yang paling banyak data hilang ke yang paling sedikit, kita bisa menentukan kolom mana yang paling perlu ditangani.

Dalam kode, kita bisa gunakan parameter ascending=False untuk membuat urutan dari yang tertinggi ke terendah.

**Penjelasan Matematis: Ekstraksi Informasi Tanggal**

**1. Mengubah Teks Tanggal ke Format Waktu**

Agar data tanggal bisa diproses lebih lanjut, langkah pertama adalah **mengonversi string menjadi objek waktu (datetime)**.  
Hal ini dilakukan menggunakan fungsi bawaan pandas pd.to\_datetime(), yang secara otomatis akan mencoba mem-parsing format tanggal.

Jika ada nilai yang tidak bisa dikonversi (misalnya formatnya aneh atau kosong), parameter errors='coerce' akan secara otomatis menggantinya dengan nilai khusus bernama NaT (Not a Time) — semacam versi waktu dari NaN.

Secara konsep, proses ini bisa dianggap sebagai fungsi transformasi dari domain string → domain waktu:

A black background with white text

AI-generated content may be incorrect.

**2. Mengambil Komponen-Kompnen dari Tanggal**

Setelah format tanggal berhasil dibentuk, kita bisa **mengekstrak elemen-elemen pentingnya**, seperti tahun, bulan, hingga nama bulan.  
Ini dilakukan lewat properti .dt dari pandas, yang membuka akses ke atribut waktu.

Beberapa ekstraksi umum antara lain:

* **Tahun** → dt.year: Menghasilkan angka 4 digit (misalnya 2020)
* **Bulan** → dt.month: Memberikan angka 1–12 sesuai urutan bulan
* **Nama Bulan** → dt.strftime('%B'): Mengembalikan nama lengkap bulannya (misal: "March", "September")

**3. Dari Satu Kolom Menjadi Banyak: Feature Expansion**

Proses ini mengubah satu buah kolom waktu (date\_added) menjadi beberapa fitur terpisah seperti:

* Tahun penambahan
* Nomor bulan
* Nama bulan

Proses seperti ini disebut **feature expansion**, bagian dari teknik *feature engineering* yang sangat umum dilakukan di tahap preprocessing.

Secara sederhana, jika kita anggap kolom tanggal adalah x, maka:

A black and white logo

AI-generated content may be incorrect.

**Penjelasan Matematis: Ekstraksi Nilai dan Jenis Durasi**

**1. Ekstraksi Nilai Durasi (Angka)**

Fungsi pertama, **ekstrak\_nilai\_durasi**, bertugas untuk **mengekstrak nilai numerik** dari string yang berisi informasi durasi (seperti "1 Season" atau "90 Minutes").

Langkah-langkahnya:

* **Langkah pertama**: Fungsi ini mengecek apakah nilai durasi\_str adalah NaN (missing value). Jika iya, fungsi akan mengembalikan NaN.
* **Langkah kedua**: Fungsi menggunakan **regular expression** \d+ untuk mencari angka yang ada dalam string. Regular expression \d+ mencari urutan digit berturut-turut di dalam string.
* **Langkah ketiga**: Jika angka ditemukan, angka tersebut dikonversi menjadi tipe data integer. Jika tidak ada angka yang ditemukan, fungsi mengembalikan NaN.

Secara matematis, ini adalah **pemrosesan string untuk mengekstraksi komponen numerik** dari sebuah teks.

A black background with white text

AI-generated content may be incorrect.

**2. Ekstraksi Jenis Durasi (Satuan)**

Fungsi kedua, **ekstrak\_jenis\_durasi**, bertugas untuk **menentukan unit waktu** dari durasi tersebut, apakah dalam satuan "Season" atau "Minute".

* **Langkah pertama**: Sama seperti fungsi pertama, ia memeriksa apakah nilai durasi\_str adalah NaN. Jika ya, mengembalikan NaN.
* **Langkah kedua**: Fungsi memeriksa apakah string mengandung kata 'Season'. Jika ada, maka unitnya adalah 'Season', jika tidak, maka diambil 'Minute'.

Secara matematis, ini adalah fungsi **pemetaan kategori** yang memilih antara dua opsi berdasarkan kondisi.

A black background with white text

AI-generated content may be incorrect.

**Ekspansi Fitur (Feature Expansion)**

* **Transformasi 1 Kolom Menjadi Banyak Kolom**:  
  Dengan dua fungsi ekstraksi di atas, satu kolom duration yang berisi teks, diproses menjadi dua kolom baru: nilai\_durasi (angka) dan tipe\_durasi (jenis satuan). Ini adalah contoh dari **feature expansion**, di mana satu fitur (kolom) dipecah menjadi beberapa fitur baru untuk analisis lebih lanjut.

Secara matematis, proses ini dapat digambarkan sebagai fungsi yang mentransformasikan satu fitur menjadi dua fitur baru:



**Penjelasan Matematis: Ekstraksi Jumlah Director dan Cast**

1. **Fungsi Pemetaan**:  
   Fungsi hitung\_sutradara\_unik dan hitung\_pemeran\_unik melakukan hal yang sama—memecah string menjadi nama individu, membersihkan spasi ekstra dengan strip(), dan menghapus entri duplikat menggunakan set().

Secara matematis, kita dapat menuliskannya sebagai:

1. **Set dan Strip**:  
   Penggunaan set() memastikan bahwa hanya nama unik yang dihitung, dan strip() memastikan tidak ada nama yang dihitung dengan spasi yang tidak perlu.

**Penjelasan Matematis: Ekstraksi Jumlah Genre dan Negara**

1. **Pemisahan Elemen Berdasarkan Koma: Untuk setiap nilai dalam kolom, kita melakukan pemisahan berdasarkan koma, sehingga kita mendapatkan set elemen. Jumlah elemen tersebut adalah hasil dari fungsi tersebut. Secara matematis, dapat dijelaskan sebagai:**

****

1. **Penghitungan Jumlah Elemen: Setelah data dipisahkan berdasarkan koma, kita hanya menghitung berapa banyak elemen (genre atau negara) yang terpisah. Ini adalah operasi counting dari elemen-elemen yang valid.**

**Penjelasan Matematis: Perhitungan Umur Konten**

Fungsi untuk menghitung umur konten ini didasarkan pada rumus sederhana:



Ini adalah operasi pengurangan antara tahun saat ini (tahun\_aktual) dengan tahun rilis (release\_year) untuk mendapatkan usia konten dalam bentuk tahun.

**Penjelasan Matematis: Klasifikasi Rating**

Pemetaan ini adalah contoh dari fungsi transformasi diskret:

A white arrow pointing to the left

AI-generated content may be incorrect.

Dimana fungsi f didefinisikan sebagai:



Alih-alih menggunakan banyak if, pemetaan dilakukan melalui struktur data dict, yang secara komputasi juga lebih efisien.

**Exploratory Data Analysis (EDA)**

**Penjelasan Matematis: Distribusi Kategorikal**

**Dalam konteks ini, kita sedang menganalisis distribusi frekuensi dari variabel kategorikal type yang hanya memiliki dua kemungkinan: "Movie" atau "TV Show".**

**Secara matematis:**

* **Kita ingin menghitung:**

**A black background with white numbers and symbols

AI-generated content may be incorrect.**

Di mana:

* Xi adalah isi dari kolom type pada baris ke-i
* c adalah kategori tertentu (misalnya Movie)
* 1 adalah **fungsi indikator** yang bernilai 1 jika kondisi benar, dan 0 jika salah

Hasil dari operasi ini adalah:

* Total konten Movie dan TV Show
* Visualisasi digunakan untuk membandingkan skala kategori secara visual

**Penjelasan Matematis: Distribusi Rating**

**menggunakan groupby() untuk mengelompokkan data berdasarkan dua variabel: type dan rating, kemudian kita menghitung jumlah elemen di setiap kelompok menggunakan size(). Fungsi reset\_index() digunakan untuk merubah hasil grup menjadi DataFrame kembali.**

**Secara matematis:**

**A black background with white text

AI-generated content may be incorrect.**

* **xi adalah rating pada baris ke-i.**
* **yi​ adalah tipe konten pada baris ke-i.**
* **r adalah rating tertentu (misalnya 'PG', 'R', 'TV-MA').**
* **t adalah tipe konten ('Movie' atau 'TV Show').**

**countplot() digunakan untuk visualisasi distribusi rating, dengan hue digunakan untuk membedakan konten berdasarkan tipe (Movie vs TV Show). Dengan ini, kita dapat melihat distribusi jumlah konten berdasarkan rating dan tipe konten.**

**Penjelasan matematis trend pembuatan konten:**

**1. Ekstraksi Tahun dari date\_added**

**Proses ekstraksi tahun dari kolom date\_added adalah langkah pertama dalam analisis tren penambahan konten per tahun. Secara matematis, proses ini bisa dianggap sebagai operasi ekstraksi komponen waktu dari objek datetime.**

* **Ekstraksi tahun:**
  + **Misalkan date\_added adalah sebuah tanggal yang memiliki format yyyy-mm-dd (seperti 2021-07-15).**
  + **Kita ingin mengambil hanya komponen tahun dari tanggal tersebut. Secara matematis, kita melakukan operasi pemisahan tahun dari komponen lain:**

****

* + **Sebagai contoh, jika date\_added = "2021-07-15", maka hasil ekstraksi tahun adalah 2021.**

**2. Menghitung Jumlah Konten per Tahun**

**Setelah kita mengekstrak tahun dari kolom date\_added, kita dapat menghitung berapa banyak konten yang ditambahkan pada setiap tahun. Secara matematis, ini adalah perhitungan frekuensi atau count dari tahun-tahun yang ada dalam dataset. Misalnya, jika kita memiliki dataset dengan informasi tanggal penambahan, kita ingin menghitung jumlah entri per tahun.**

* **Misalkan kita memiliki daftar tahun: 2020,2020,2021,2021,2021,20222020, 2020, 2021, 2021, 2021, 20222020,2020,2021,2021,2021,2022.**
* **Frekuensi setiap tahun dapat dihitung sebagai berikut:**

****

* **Hasilnya adalah jumlah konten yang ditambahkan pada tahun-tahun tersebut.**

**3. Visualisasi Tren Penambahan Konten**

**Visualisasi tren penambahan konten menggunakan area plot atau line plot adalah representasi grafis dari frekuensi konten yang ditambahkan per tahun. Jika kita menggambarkan jumlah konten yang ditambahkan setiap tahun pada sumbu vertikal (y), dan tahun pada sumbu horizontal (x), maka kita mendapatkan kurva atau area yang menggambarkan tren.**

**Secara matematis, plot ini menggambarkan fungsi jumlah konten per tahun:**

****

**Di sini, f(x)f(x)f(x) adalah jumlah konten pada tahun xxx.**

* **Area Plot: Area di bawah kurva menggambarkan total konten yang ditambahkan pada setiap tahun. Ini adalah representasi integral dari jumlah konten untuk setiap tahun, yang menunjukkan total kumulatif dalam rentang waktu yang diberikan.**

**A black background with white text

AI-generated content may be incorrect.**

* Di sini, integral ini menunjukkan jumlah total konten yang ditambahkan dalam rentang tahun x1sampai x2​.
* **Line Plot**: Garis yang menghubungkan titik-titik frekuensi pada plot menggambarkan fluktuasi penambahan konten seiring waktu.

Secara umum, proses ini menunjukkan bagaimana konten di platform seperti Netflix berkembang seiring waktu, dan memvisualisasikan tren ini dengan menggunakan plot memungkinkan kita untuk lebih mudah memahami pola penambahan konten.

Kesimpulannya :

**1. Pembersihan dan Persiapan Data:**

* Proses pembersihan data termasuk mengatasi nilai hilang (missing values) dan memastikan kolom data yang relevan, seperti added\_year, rating\_category, duration\_unit, dan lain-lain, berada dalam format yang benar.
* Beberapa kolom baru juga diekstraksi, seperti **jumlah sutradara, pemeran, genre, negara**, dan **umur konten**, yang memungkinkan analisis lebih lanjut untuk melihat hubungan antar variabel.

**2. Eksplorasi Data:**

* **Distribusi Tipe Konten**: Analisis jenis konten (Movie vs TV Show) menunjukkan bagaimana kedua kategori ini tersebar dalam dataset berdasarkan tahun dan jumlah konten yang ditambahkan.
* **Distribusi Rating**: Menampilkan distribusi rating konten, dengan analisis visual seperti **histogram** dan **countplot**, membantu memahami kategori rating yang paling umum untuk konten Netflix.
* **Distribusi Genre dan Negara**: Mengekstrak dan menganalisis jumlah genre dan negara per konten untuk mengetahui sejauh mana variasi genre dan lokasi geografis dapat ditemukan dalam dataset.

**3. Transformasi dan Feature Engineering:**

* Berbagai transformasi dilakukan untuk membuat fitur baru seperti **jumlah sutradara**, **jumlah pemeran**, **jumlah genre**, **jumlah negara**, dan **umur konten**. Ini memberikan wawasan lebih dalam mengenai karakteristik konten di Netflix.
* **Kategorisasi Rating**: Rating dikategorikan dalam kelompok usia yang relevan, seperti "Kids", "Teens", dan "Adults", untuk memahami lebih baik jenis audiens yang dimaksudkan oleh setiap rating.

**4. Visualisasi dan Tren:**

* Berbagai visualisasi digunakan untuk menyajikan distribusi, hubungan, dan tren dalam dataset, seperti:
  + **Distribusi Rating** dengan **countplot** dan **word cloud** untuk melihat frekuensi rating dan popularitas kata dalam judul dan deskripsi.
  + **Boxplot** untuk memeriksa distribusi umur konten per kategori rating, memberikan gambaran bagaimana umur konten bervariasi di berbagai rating.
  + **Tren Penambahan Konten per Tahun**: Analisis jumlah konten yang ditambahkan setiap tahun untuk melihat apakah ada tren signifikan dalam penambahan Movie vs TV Show.

**5. Korelasi antar Fitur:**

* Korelasi antara fitur numerik menunjukkan hubungan antar variabel, yang bisa digunakan untuk analisis lebih lanjut dan model prediksi. Matrix korelasi membantu mengidentifikasi fitur-fitur yang berhubungan erat, yang penting dalam analisis lebih mendalam.

**6. Distribusi dan Tren Konten:**

* Analisis distribusi **jumlah season** untuk TV Show memberikan wawasan tentang panjang serial yang tersedia di Netflix, sementara **jumlah cast** memberi gambaran tentang besarnya produksi yang terlibat dalam pembuatan konten.
* **Distribusi Penambahan Konten per Bulan** memberikan wawasan tentang periode-periode puncak dalam penambahan konten, yang mungkin terkait dengan strategi penjadwalan atau pemasaran Netflix.