**Rangkuman Data Visualization**

**Pengenalan Data Visualization**

Pada siklus tersebut, data visualization merupakan tahapan yang harus kita lakukan sebelum membuat kesimpulan dan mengomunikasikan (*draw conclusion* & *communicate*) hasil dari proses analisis yang telah dilakukan. Hal ini dilakukan untuk mempermudah kita dalam memahami data dan membuat sebuah kesimpulan yang andal. Selain itu, visualisasi data juga dapat membantu kita dalam menyampaikan cerita atau temuan dari hasil analisis data kepada orang lain atau stakeholder.

Sederhananya, visualisasi data merupakan cara kita dalam menyajikan data dalam bentuk visual. Hal ini dilakukan untuk mempermudah kita dan orang lain dalam memahami data tersebut. Selain itu, visualisasi data yang baik juga akan sangat membantu kita dalam menyampaikan story dan pesan dari sebuah data.

**Prinsip-Prinsip dalam Visualisasi Data**

Ketika berbicara tentang visualisasi data, kita tidak hanya membahas cara menyajikan data ke dalam bentuk visual, melainkan juga harus memahami berbagai prinsip dasar dalam membuat visualisasi data yang baik dan benar.

Umumnya visualisasi data yang buruk memuat salah satu dari tiga masalah berikut.

* **Misleading**  
  Masalah pertama dalam visualisasi data yang buruk ialah *misleading information* yang mampu mengakibatkan kesalahan dalam pengambilan kesimpulan dari sebuah data.
* **Hides**  
  Selain *misleadning information*, masalah lain yang umum dijumpai dalam visualisasi data yang buruk ialah menyembunyikan informasi tertentu.
* **Distracts**  
  Masalah lain yang mungkin terjadi ialah distraksi. Visualisasi data yang buruk sering kali menyertakan komponen visual yang seharusnya tidak dibutuhkan dan malah mengganggu proses pengambilan kesimpulan dari sebuah visualisasi data.

Jika ditelisik secara mendasar, visualisasi data merupakan proses dalam mengubah data ke dalam bentuk visual menggunakan berbagai elemen visual. Berikut merupakan beberapa elemen visual yang umum digunakan untuk membuat visualisasi data.

* **Position**: elemen ini akan membantu kita merepresentasikan titik data menggunakan sumbu tertentu (seperti sumbu X, Y, dan Z) sebagai acuan.
* **Size**: ukuran (panjang atau lebar) merupakan elemen visual yang umumnya kita gunakan untuk membedakan serta membandingkan nilai dari kategori atau titik data tertentu.
* **Shape**: bentuk merupakan salah satu elemen visual yang dapat digunakan untuk membedakan kategori atau titik data tertentu.
* **Color**: selain bentuk, warna juga merupakan pilihan elemen visual lain yang dapat digunakan untuk membedakan kategori atau titik data tertentu. Ketika menggunakan elemen ini, kita harus ingat bahwa tidak semua orang memiliki kemampuan untuk membedakan warna dengan baik.
* **Texture**: penambahan tekstur atau pola tertentu bisa menjadi alternatif lain dalam membedakan kategori atau titik data tertentu.
* **Angle**: pada beberapa pilihan bentuk visualisasi data, sudut merupakan salah satu elemen visual yang digunakan untuk merepresentasikan nilai dari suatu data.

Untuk memastikan penggunaan elemen visual secara efektif, Edward Tufte telah mengenalkan sebuah konsep bernama **Data Ink Ratio**pada bukunya yang berjudul “*The Visual Display of Quantitative Data*”.

Data ink ratio merupakan sebuah konsep parameter yang digunakan sebagai penentu efektivitas penggunaan visual elemen dalam sebuah visualisasi data. Ia didefinisikan sebagai perbandingan antara tinta (bisa diartikan sebagai elemen visual) yang digunakan untuk mendeskripsikan data dan total tinta yang digunakan dalam satu visualisasi data.

Ketika membuat sebuah visualisasi data, kita tidak hanya memikirkan terkait faktor estetika (seni) saja. Melainkan, kita juga harus memastikan visualisasi data yang kita buat mampu menyampaikan berbagai informasi penting dari sebuah data tanpa menyembunyikan fakta, mengarahkan orang lain pada asumsi tertentu, serta menghindari *misinterpretasi*.

Pada bukunya yang berjudul The Visual Display of Quantitative Data, Edward Tufte juga mengenalkan beberapa prinsip guna menjaga integritas dalam membuat visualisasi data. Berikut beberapa prinsip tersebut.

* Representasi angka pada sebuah grafik harus sebanding dengan kuantitas sebenarnya. Hal ini untuk menghindari terjadinya misinterpretasi dari sebuah visualisasi data.
* Pelabelan yang jelas, terperinci, dan menyeluruh merupakan kunci utama dalam menghindari ambiguitas dalam sebuah visualisasi data.
* Visualisasi data yang baik harus menunjukkan variasi pada data bukan variasi desain.
* Untuk menghindari asumsi, kita harus menggunakan unit yang umum digunakan (sesuai standar atau kesepakatan) dalam merepresentasikan suatu data tertentu.
* Jumlah variabel (*feature*/kolom) yang digambarkan tidak boleh melebihi jumlah dimensi dalam data. Sebagai contoh, untuk menggambarkan hubungan variabel X dan Y, kita disarankan untuk menggunakan plot dua dimensi ketimbang menggunakan plot tiga dimensi.
* Sebuah grafik tidak boleh menampilkan data yang di luar konteks.

Nah, itu dia keenam prinsip yang harus kita ikuti untuk menjaga integritas dan menghindari misinterpretasi dalam sebuah visualisasi data. Sebenarnya pada buku yang sama, Edward Tufte juga mengenalkan sebuah parameter bernama **Lie Factor** untuk mengukur kesalahan representasi dalam sebuah visualisasi data. Parameter ini didefinisikan sebagai rasio perbandingan ukuran yang ditampilkan pada grafik dan ukuran yang sebenarnya ada dalam data. Rasio ini tentunya akan memberikan kita gambaran sejauh mana visualisasi mendistorsi nilai dari sebuah data.

**Univariate Visualization**

Seperti yang telah kita bahas sebelumnya, visualisasi data merupakan teknik yang digunakan dalam merepresentasikan atau menyajikan data ke dalam bentuk visual yang lebih mudah dipahami. Nah, berdasarkan jumlah variabel (*feature*/kolom) yang ingin disajikan, visualisasi data dapat dibagi menjadi tiga kategori.

* **Univariate visualization**  
  Univariate visualization merupakan bentuk visualisasi data yang hanya merepresentasikan informasi yang terdapat pada satu variabel. Jenis visualisasi ini umumnya digunakan untuk memberikan kita gambaran terkait distribusi sebuah variabel dalam suatu dataset.
* **Bivariate visualization**  
  Bivariate visualization merupakan bentuk visualisasi data untuk menyajikan informasi yang terdapat dalam dua variabel. Visualisasi jenis ini dapat digunakan untuk menganalisis hubungan antar dua variabel dalam sebuah dataset.
* **Multivariate visualization**  
  *Last but not least*, multivariate visualization merupakan jenis visualisasi data untuk menggambarkan informasi yang terdapat dalam lebih dari dua variabel. Jenis visualisasi ini digunakan untuk merepresentasikan hubungan dan pola yang terdapat dalam multidimensional data.

**Bar Chart**

Bentuk visualisasi data pertama yang akan kita bahas ialah bar chart atau dikenal juga dengan sebutan diagram batang. Ia merupakan pilihan visualisasi data yang digunakan untuk menggambarkan distribusi dari suatu data kategoris (*categorical data*).

Pada bar chart, setiap kategori akan direpresentasikan sebagai sebuah batang. Tinggi dari sebuah batang menunjukkan frekuensi atau jumlah titik data dari kategori tersebut.

Library matplotlib telah menyediakan sebuah function bernama **pyplot.bar()** untuk membuat sebuah bar chart. Untuk menggunakan function tersebut, kita perlu mendefinisikan parameter **x** dan **height** dari bar chart yang ingin dibuat.

Selain menggunakan library matplotlib, kita juga bisa membuat bar chart menggunakan library seaborn. Untuk membuat bar chart menggunakan library seaborn, kita bisa menggunakan function **barplot()**. Function ini akan menyediakan beberapa parameter penting seperti berikut.

* **data**: menampung DataFrame yang akan digunakan.
* **x, y**: menampung nama kolom atau data yang divisualisasikan.
* **orient**: orientasi dari bar chart yang akan digunakan (“v” atau “h”).
* **color**: mendefinisikan warna yang akan digunakan.

**Pie Chart**

Pie chart merupakan bentuk visualisasi data yang dapat digunakan untuk menggambarkan perbandingan frekuensi tiap kategori pada suatu data kategoris. Pada pie chart, setiap kategori digambarkan oleh irisan dalam sebuah lingkaran dan frekuensi digambarkan sebagai luasan (*area*) dari irisan tersebut.

Terdapat variasi lain dari pie chart, yaitu donut plot. Sebenarnya, ia merupakan pie chart yang memiliki lubang pada bagian tengahnya sehingga membentuk donut. Selain itu, tidak ada perbedaan mendasar dalam penggunaan antara pie chart dan donut plot.

Untuk membuat pie chart, kita bisa memanfaatkan library andalan, yaitu matplotlib. Ia telah menyediakan sebuah function bernama **pie()** untuk menghasilkan grafik pie chart. Function tersebut memiliki beberapa parameter penting yaitu seperti berikut.

* **x**: menampung data yang akan divisualisasikan.
* **explode**: menampung array atau list yang mengatur posisi tiap irisan lingkaran.
* **labels**: sekumpulan string yang digunakan untuk memberi label pada tiap irisan lingkaran.
* **colors**: sekumpulan warna yang akan digunakan pada tiap irisan lingkaran.
* **autopct**: string yang digunakan untuk memberi numerik label pada tiap irisan lingkaran.

**Histogram**

Jika dilihat sekilas, bentuk grafik histogram mirip dengan bentuk bar chart perbedaannya hanya terdapat pada sumbu X. Pada grafik histogram, sumbu X digunakan untuk menampung range nilai dari data kuantitatif yang dikenal dengan istilah *bins*.

Pada grafik histogram, ukuran bins akan sangat berpengaruh terhadap cara kita dalam merepresentasikan data kuantitatif. Jika menggunakan ukuran bins yang terlalu besar, kita akan kehilangan cukup banyak informasi dari data kuantitatif yang ingin divisualisasikan. Sebaliknya, apabila ukuran bins yang digunakan terlalu kecil, hal ini akan menghasilkan banyak noise yang mendistraksi distribusi data yang ingin kita amati.

Untuk membuat grafik histogram, kita bisa menggunakan function **hist()**yang disediakan oleh library matplotlib. Function ini menerima beberapa parameter seperti berikut.

* **x**: menampung data yang akan divisualisasikan.
* **bins**: menampung jumlah bins (sebanding dengan ukurannya) yang akan digunakan untuk membuat grafik histogram.

Selain menggunakan library matplotlib, kita juga bisa membuat grafik histogram dengan library seaborn. Library ini menyediakan function **histplot()** untuk membuat grafik histogram.

**Box Plot**

Bentuk visualisasi data lain yang bisa digunakan untuk mengidentifikasi distribusi pada data kuantitatif ialah box plot. Ia merupakan bentuk visual untuk merepresentasikan nilai IQR beserta ambang batas bawah dan atas dari sebuah data. Box plot ini juga bisa digunakan untuk mengidentifikasi outlier atau pencilan yang ada pada data kuantitatif. Kita bisa menggunakan function**boxplot()** yang disediakan oleh library seaborn membuat plot jenis ini.

**Bivariate dan Multivariate Visualization**

Sederhananya, bivariate visualization merupakan bentuk visualisasi data untuk menyajikan informasi dari dua variabel dalam suatu dataset. Di lain sisi, multivariate visualization merupakan jenis visualisasi data untuk menggambarkan informasi yang terdapat dalam lebih dari dua variabel. Keduanya sering digunakan untuk mendeskripsikan hubungan antar variabel dalam suatu dataset.

**Scatter Plot**

Scatter plot merupakan bentuk visualisasi data yang digunakan untuk melihat hubungan antara dua atau lebih variabel data kuantitatif. Pada scatter plot sumbu X, Y, atau Z digunakan untuk menampung nilai dari setiap variabel yang akan divisualkan. Titik pada scatter plot merupakan titik temu data dari setiap sumbu yang digunakan.

Untuk membuat scatter plot, kita bisa menggunakan library matplotlib. Ia telah menyediakan sebuah function bernama **scatter()** untuk membantu kita membuat scatter plot secara lebih mudah.

Selain menggunakan library matplotlib, kita juga bisa menggunakan function **scatterplot()** yang disediakan oleh library seaborn untuk membuat scatter plot.

Untuk mempermudah dalam melihat korelasi atau hubungan antar variabel, kita bisa menggunakan function **regplot()** yang disediakan oleh library seaborn. Function tersebut memadukan scatter plot dan*regression function* (metode statistik untuk memperkirakan korelasi antar *independent* dan *dependent variable*) untuk melihat tren serta korelasi antar variabel.

**Line Chart**

Line chart merupakan bentuk grafik yang umum digunakan untuk menggambarkan tren nilai dari suatu variabel terhadap variabel lain. Kita bisa membuat line chart menggunakan function **plot()** yang disediakan oleh matplotlib.

Bentuk visualisasi data ini umumnya digunakan untuk melihat trend dari data berbentuk *time series* (data yang direkam dalam interval waktu yang konsisten).

**Clustered Bar Chart**

Clustered bar chart merupakan bentuk modifikasi dari bar chart yang sebelumnya kita kenal dengan menambahkan variabel kategoris lain. Modifikasi ini memungkinkan untuk melihat distribusi serta hubungan antar dua atau lebih variabel kategoris.

Untuk membuat clustered bar chart, kita bisa menggunakan parameter **hue** pada function **barplot()** yang disediakan oleh library seaborn.

**Menggunakan Komponen Visual Tambahan**

Selain menggunakan scatter plot, line chart, dan clustered bar chart, kita juga bisa memodifikasi bentuk visualisasi data dengan menambahkan komponen visual tambahan seperti *color*, *shape,* serta *texture*untuk mendeskripsikan lebih banyak variabel.

Sebagai contoh, Anda bisa membuat lebih dari satu boxplot untuk menggambarkan distribusi nilai numerik dari beberapa variabel. Hal ini bisa Anda gunakan dengan memasukkan DataFrame yang akan divisualisasikan pada parameter **data**dalam function **boxplot()**. Selain itu, Anda juga bisa menggunakan parameter **palette** untuk mengatur palet warna yang akan digunakan.

**Explanatory Analysis**

Explanatory analysis merupakan proses analisis data yang bertujuan untuk membagikan beberapa *insight* yang menarik dari sebuah data. Insight yang diperoleh dari proses explanatory analysis akan digunakan untuk menjawab permasalahan bisnis yang dihadapi. Insight tersebut juga akan dibagikan kepada orang lain (*stakeholder*) sehingga perlu disertakan dengan visualisasi data yang baik

Pada proses explanatory analysis, visualisasi data digunakan sebagai media untuk menyampaikan pesan atau *story* dari sebuah data. Oleh karena itu, kita harus menyampaikan informasi terkait konteks dari sebuah visualisasi data. Hal ini bisa dicapai dengan menambahkan beberapa komponen berikut.

* **Title**  
  Title atau judul merupakan salah satu komponen penting guna menyampaikan konteks pada sebuah visualisasi data. Kita bisa menambahkan judul pada visualisasi data menggunakan function **title()** yang disediakan oleh matplotlib.
* **Label axes**  
  Selain judul visualisasi, kita juga perlu menambahkan label atau keterangan pada setiap sumbu yang terdapat dalam visualisasi data. Hal ini dilakukan untuk memberikan konteks terkait titik data yang direpresentasikan. Kita bisa membuat label axes menggunakan function **xlabel()** dan **ylabel()** yang disediakan oleh library matplotlib.
* **Legend**  
  Pastikan Anda menambahkan legend memberikan keterangan untuk variabel yang tidak digambarkan pada sumbu. Keterangan yang digunakan akan bergantung pada cara kita dalam merepresentasikan variabel tambahan tersebut. Sebagai contoh, kita bisa menggunakan **legend()** untuk memberikan keterangan pada variabel kategoris tambahan. Namun, apabila variabel tambahan tersebut berupa nilai numerik yang direpresentasikan menggunakan range warna, kita bisa memanfaatkan function **colorbar()** untuk memberikan keterangan pada variabel tersebut.

Selain informasi konteks, pastikan audiens fokus pada informasi yang ingin kita sampaikan. Oleh karena itu, sebagai seorang praktisi data yang andal, kita harus mampu mengarahkan perhatian audiens. Untuk melakukan hal ini, kita bisa menggunakan komponen visual tertentu seperti *color*, *shape*, dan *texture*.

Sebagai contoh, kita bisa menggunakan warna yang kontras untuk mengarahkan perhatian audiens pada bagian yang ingin di-*highlight*.

Namun, perlu diingat bahwa ketika menggunakan teknik ini kita harus menghindari penggunaan warna yang menyakitkan mata audiens atau warna yang memiliki arti berlawanan dengan pesan yang ingin disampaikan. Selain itu, kita juga memperhatikan konsistensi dalam pemilihan warna dalam seluruh visualisasi data yang kita buat.

Pada proses pembuatan explanatory analysis, terkadang sulit untuk menilai seberapa efektif visualisasi data yang kita buat. Oleh karena itu, kita membutuhkan feedback serta sudut pandang orang lain guna memastikan bahwa visualisasi data yang kita buat sudah cukup efektif dalam menyampaikan story dan informasi penting dari sebuah data.

Pada proses ini kita bisa meminta teman atau rekan setim untuk melihat seluruh visualisasi data yang telah kita buat. Selanjutnya, mintalah mereka untuk membuat kesimpulan dari seluruh visualisasi data yang telah dibuat. Lalu, pastikan kesimpulan mereka sesuai dengan yang ingin Anda sampaikan. Pada prosesnya, Anda bisa mengajukan beberapa pertanyaan berikut.

* Informasi apa yang Anda peroleh dari setiap visualisasi data tersebut?
* Menurut Anda apa kesimpulan utama dari seluruh visualisasi data tersebut?
* Apakah Anda memiliki pertanyaan terkait data tersebut yang belum terjawab?
* Apakah ada bagian yang belum Anda pahami dari seluruh visualisasi data tersebut?
* Berapa lama waktu yang Anda butuhkan untuk memahami setiap visualisasi data tersebut?

Anda dapat menggunakan beberapa pertanyaan tersebut guna mendapatkan feedback atau masukan untuk meningkatkan kualitas visualisasi data yang dibuat. Selain itu, pertanyaan tersebut juga membantu Anda mengidentifikasi informasi penting lain yang mungkin Anda lewatkan.