데이터와 시각화

데이터파트

데이터를 global하게 보느냐 local하게 보느냐

데이터를 어디서 얻지?

캐글

정형 데이터

Csv, tsv파일로 제공됨/ row가 한 아이템, col이 attribute(feature)

시계열 데이터(time-series)

시간흐름이 존재하면 이에 해당/ 추세 계절성 주기성 가짐

지리/지도 데이터

지도 정보와 보고자 하는 정보간의 조화 중요/거리 경로 분포 등 다양한 실사용

위도 경도와 같은 정보 중요

관계 데이터

객체와 객체 간의 관계를 시각화

객체는 node, 관계는 link로 표현

Json, 2darray, linkedlist 등으로 데이터를 받을수 있다

가독성이 좋진 않아서 맵핑 방법이 중요하다

계층적 데이터

관계 중에서도 포함관계가 분명한 데이터

Tree, treemap, sunburst 등이 대표적이다

Numerical/ categorical data

Continuous,discrete -------- nominal,ordinal

수치적인 통계가 중요하면 이산형으로

마크와 채널

Pre-attentive attribute

주의를 주지 않아도 인지하게되는 요소

그냥 보면 다른게 보이는 요소

동시에 사용하면 인지하기 어려움

Visual pop-out 시각적 분리

bar barh

stacked bar plot

쌓아서 표현

그룹의 순서는 항상 유지/ 맨 및의 bar의 분포는 파악하기 쉽지만 그 외는 no

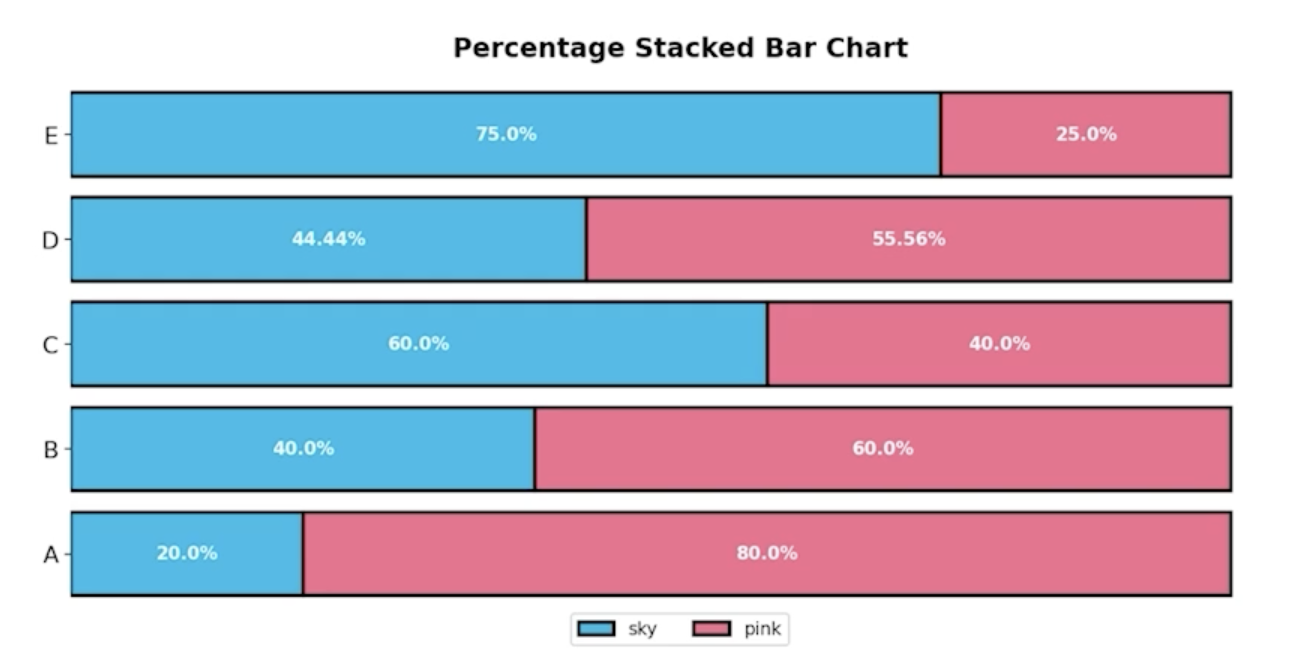
2그룹이 positive/negative라면 축 조정 가능

.bar(). Bottom

Barh(). Left

Percentage stacked bar chart

전체에서 비율을 나타냄



Overlapped bar plot

2개 그룹만 비교한다면 겹쳐서 만드는것도 하나의 선택지

같은 축을 사용하니 비교가 쉬움/ 투명도 옵션 alpha

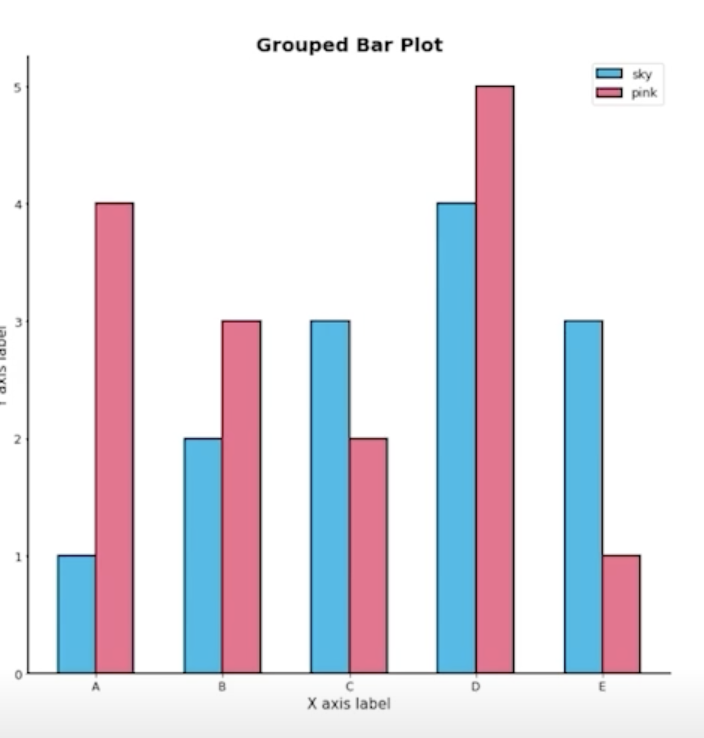
Bar plot 보다는 Area plot에서 더 효과적(seaborn에서 다룸)

가장 추천

Grouped bar plot

수치화해야해서 matplotlib에서는 구현이까다로움/ 다만 seaborn에는 만들어져있음

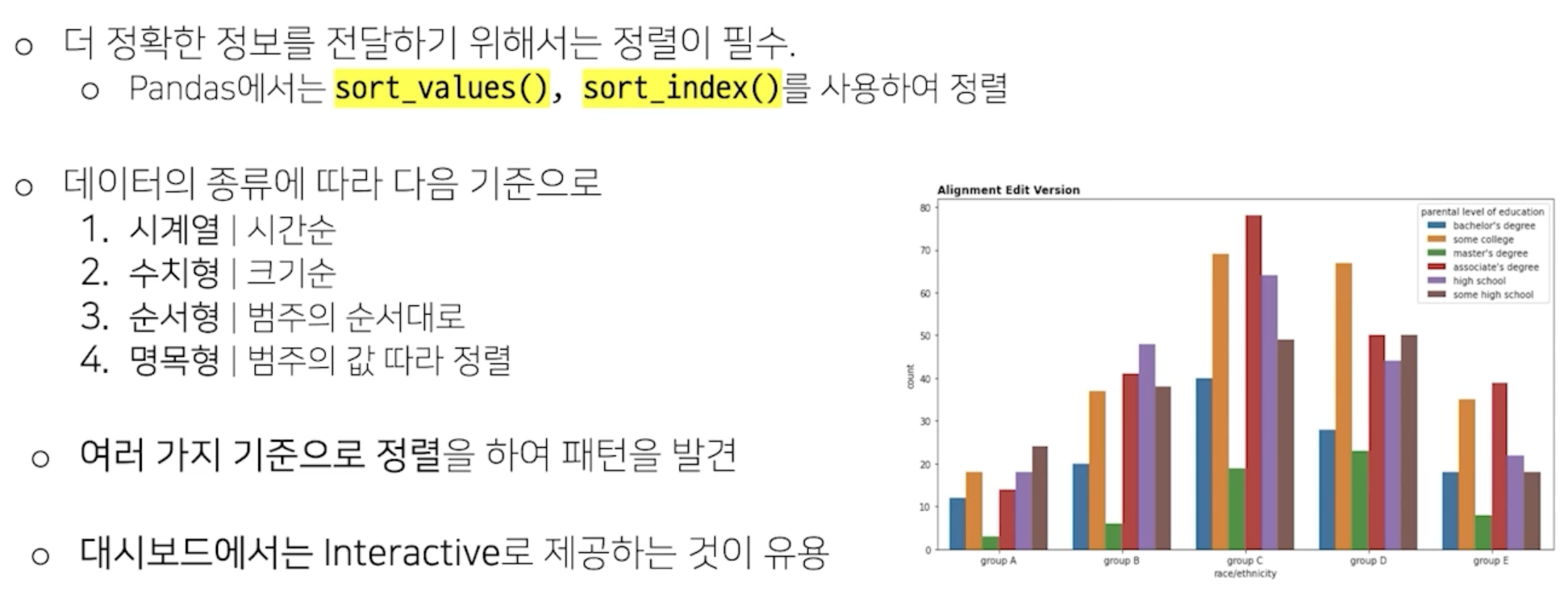
그룹이 5개에서 7개 이하일때 효과적/ 그 이상이면 적은 그룹은 etc로 처리



반드시 x축의 시작은 0

만약 차이를 나타내고 싶으면 plot의 세로 비율을 늘리기

1과 5를 표현한다면 각각에 필요한 잉크 양은 1과 5여야 한다.



여백과 공간만 조정해도 가독성이 높아진다.

텍스트이(가) 표시된 사진

자동 생성된 설명

3차원으로 하고 싶으면 interactive이용

직사각형이 아닌 다른 형태의 bar는 지양

오차막대 추가 errorbar

바 사이의 갭이 0이라면 히스토그램을 사용

.hist()사용/ 연속된 느낌을 줄 수 있음

Line plot

.plot()

시계열데이터에 유리

5개이하의 선을 사용하는 것을 추천

구별요소 color marker markersize linestyle linewidth

시시각각 변동하는 데이터는 noise로 인해 패턴 및 추세 파악이 어려움

Noise의 인지적인 방해를 줄이기 위해 smoothing을 사용/moving average 기법

전처리과정이 필요하다

Bar plot과 다르게 꼭 축을 0에 둘 필요는 없다.

미니멈과 맥시멈값에 살짝 마진을 둬서 추세를 보는게 좋다.

Grid, annotate 등 모두 제거/ 마커 스파인 그런거 다 빼서 추세를 보기 더 편하게 하는게

규칙적인 간격의 데이터가 아니라면 각 관측값에 점으로 표시해서 기울기 정보의 오해를 줄이자

점과 점 사이에 데이터가 없기에 이를 잇는 방법을 보간(interpolation)이라 한다.

텍스트이(가) 표시된 사진

자동 생성된 설명

이중축은 비교적 지양하자

라인 끝 단에 레이블을 추가하면 식별에 도움(legend 보다는)

Min max는 추가해주면 도움이 될수도

보다 연한 색을 사용하여 uncertainty 표현 가능(신뢰구간, 분산 등)

Scatter plot

.scatter()

X축과 y축에 feature값을 매핑해서 사용

Color/marker/size

상관관계(양의/ 음의 상관관계/ 없음)

군집(cluster)/ 값 사이의 차이(gap in values)/ 이상치(outliers)

Overplotting

투명도조정,

지터링, (점의 위치를 약간씩 변경해서 안 겹치게 하는것)실효성 na~~

2차원 히스토그램,히트맵을 사용하여 깔끔한 시각화

contour plot 등고선그래프

색 연속적인값은 gradient로 표시하고 이산같은 경우에는 개별 색상으로

마커 거의 구별하기 힘들어서 단독사용하는거 비추천 그리고 도형마다 크기 고르지 않다

크기

텍스트이(가) 표시된 사진

자동 생성된 설명

인과관계와 상관관계는 다르다!

인과는 a to b

상관은 a와 b가 관계가 있다

인과관계는 항상 사전 정보와 함께 가정으로 제시할 것

추세선

추세선을 사용하면 scatter의 패턴을 유추할 수 있음

추세선이 2개이상되면 구별하기 힘듦 하나만 쓰자

Grid는 지양/ 사용한다면 최소한으로/색은 무채색

Categorical에서는 히트맵과 버블차트를 추천

Matplotlib에서 text

잘못된 전달에서 생기는 오해를 방지할 수 있다.

Title/Label/Tick Label/Legend/Annotation

Zorder 맨앞으로 가져오기 맨뒤로 가져오기…

Color

가장 중요한 것은 전하고싶은 내용을 모두 전달했는가 + 그 과정에 오해는 없었는가

이미 사용하는 색에는 이유가 있다

Categorical(범주형)

Discrete,Qualitative

독립된 색상으로 구성되어 범주형 변수에 사용

최대 10개의 색상까지 사용하며 그 외에는 기타로 묶을 것

색의 차이로 구분하는것이 특징

채도와 명도를 만지는건 지양

연속형

연속적인 색상을 사용하여 값을 표현

어두운배경에서는 밝은색이

밝은 배경에서는 어두운색이 큰 값을 표현

색상은 단일 색조로 표현하는 것이 좋고

균일한 색상 변화가 중요

발산형(Diverge)

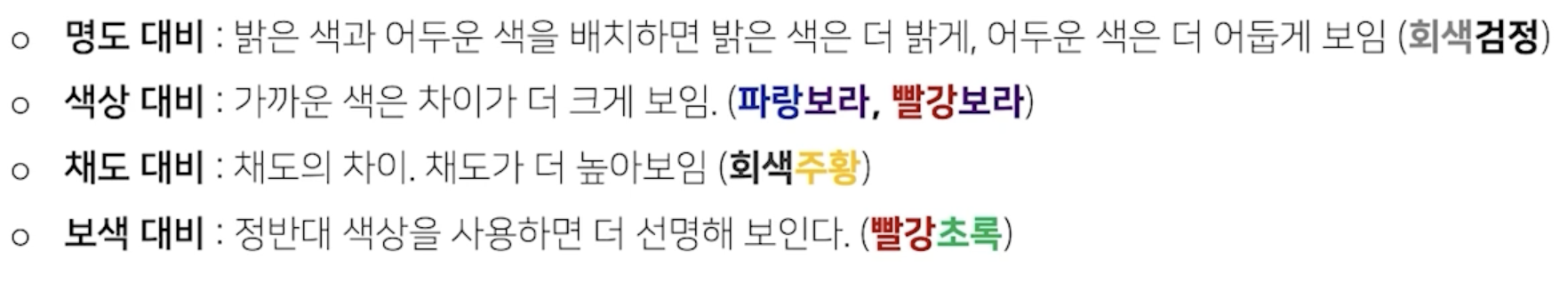
연속형과 유사하지만 중안을 기준으로 발산

상반된 값이나 서로 다른 2개를 표현하는데 적합

Ex 지지율

데이터에서 다름을 보이기위해 highlight 를 쓰기도 함

강조를 위한 방법 중 하나 -> 색상대비 사용



HSL

색조는 보면 딱 다른색인지 아는 것 빨 파 초 빨에서 보라까지 0에서 360으로 표현

채도 = 선명도

광도 = 색상의 밝기

Facet

=분할

화면상에 여러 view를 분할 및 추가하여 다양한 관점을 전달

Grid Spec

다른 크기의 subplot을 만드려면?