데이터 시각화 모범 사례

데이터 산출물을 더 읽기 쉽고, 접근하기 쉽고, 영향력 있게 만들기 위한 통찰력, 조언 및 예제(코드 포함)

Andreas Krause, Nicola Rennie, & Brian Tarran

Table of contents

1 소개 **2**

2 이 가이드 사용법 **2**
2.1 이 가이드에 기여하는 방법 . . . 3
2.2 이 가이드의 다른 사용자들과 교류하는 방법 . . . 3
2.3 저작권, 재사용, 배포 및 면책 조항 . . . 3

3 데이터 시각화를 하는 이유 **4**
3.1 동기 . . . 4
3.2 데이터 시각화의 간략한 역사 . . . 5

4 시각화의 원칙과 요소 **8**
4.1 차트의 요소 . . . 8
4.2 표의 요소 . . . 22

5 시각화 유형 선택하기 **23**
5.1 목표와 청중 . . . 23
5.2 데이터 . . . 24

6 접근성을 위한 차트 스타일링 **25**
6.1 차트 스타일링 원칙 . . . 25
6.2 색상 . . . 27
6.3 주석 . . . 36
6.4 글꼴 . . . 38
6.5 대체 텍스트 . . . 40

7 RSS 간행물을 위한 차트 스타일링 **40**
7.1 다양한 도구를 사용한 차트 스타일링 . . . 40
7.2 출판 사양 . . . 60

8 참고 문헌 **62**
8.1 가이드에서 참조된 텍스트 . . . 62
8.2 추가 읽기 자료 . . . 65
8.3 추가 리소스 . . . 65

9 저자 소개 **66**
9.1 Andreas Krause . . . 66

9.2 Nicola Rennie 67
9.3 Brian Tarran 67
9.4 기여자 67
9.5 감사의 말 68

10 이용 약관 **68**
10.1 법적 면책 조항 68
10.2 사이트 콘텐츠 69
10.3 우리가 링크하는 웹사이트는? 69
10.4 우리가 링크하지 않을 웹사이트는? 69
10.5 소프트웨어 및 서비스 69
10.6 게시 중단 정책 70
10.7 기여자 서약 행동 강령 70

1 소개

통계학은 "데이터를 수집, 분석, 제시 및 해석하는 과학"입니다(Williams, Anderson, and Sweeney 2023). 데이터 제시는 해석과 그에 따른 의사결정을 지원하고 안내하는 중요한 수단입니다. 효과적인 표시를 위한 기법들이 존재합니다. 이것이 바로 이 가이드가 다루는 내용입니다.

좋은 데이터 시각화는 데이터 제시의 기술적 측면에 대한 이해와 신중한 고려를 필요로 합니다. 하지만 여기에는 창의적인 요소도 포함됩니다. 저자는 전달하고자 하는 "이야기"에 대해 선택을 하고, 청중에게 그 이야기를 가장 효과적으로 전달할 필요성에 따라 디자인을 결정합니다. 소프트웨어 시스템은 대부분의 그래픽 요소에 대해 기본 설정을 사용합니다. 그러나 각 시각화에는 고유한 이야기가 있으므로, 작성 중인 시각화에 대한 설정을 적극적으로 고려하고 선택해야 합니다.

이 가이드는 데이터 시각화의 예술과 과학이라는 두 가지 측면을 모두 다룹니다. 주로 영국 왕립 통계학회 간행물 – 주로 Significance 매거진, Journal of the Royal Statistical Society Series A, Real World Data Science – 기여자를 위해 작성되었지만, 여기에 담긴 정보와 조언이 어떤 데이터 시각화 작업에도 광범위하게 관련되고 유용할 것이라 믿습니다.

이 가이드의 가장 중요한 목표는 독자들에게 고품질이고, 가독성이 좋으며, 정보를 효과적으로 전달하고, 표시 및 해석이 정확하며, 의도한 목적을 달성하는 데이터 시각화를 만드는 데 필요한 기초를 갖추게 하는 것입니다. 이 가이드는 우리가 데이터를 시각화하는 이유에 대한 개요로 시작하여, 차트와 표의 구조, 그리고 가독성을 높이기 위해 이러한 구조를 개선하는 방법을 포함한 데이터 시각화의 핵심 원칙과 요소를 논의합니다. 차트 스타일링을 개선하는 데 도움이 되는 구체적인 조언, 예제 및 코드가 제시되며, 특히 접근성에 중점을 둡니다. RSS 간행물을 위한 차트 스타일링에 대한 전용 섹션이 있으며, 독자들은 당면한 데이터에 적합한 차트 유형을 선택하기 위한 리소스 링크도 찾을 수 있습니다.

이 가이드를 구성하면서 저자는 해당 분야의 전문가들이 만든 많은 훌륭한 교과서, 논문 및 기타 자료를 참고했습니다. 참고문헌은 전체에 걸쳐 찾아볼 수 있으며 전용 페이지에도 수집되어 있습니다. 독자들은 데이터 시각화의 예술과 과학에 대한 이해와 감상을 깊게 하기 위해 원본 소스를 찾아볼 것을 권장합니다.

2 이 가이드 사용법

우리는 이 가이드가 처음부터 끝까지 읽히도록 설계하여, 각 섹션이 이전 섹션의 토대 위에 구축되도록 했습니다. 가이드를 처음 읽는다면 이 방식대로 읽을 것을 권장합니다. 하지만 데이터 시각화의 특정 측면에 대한 지침을 찾는 사람들은 이 웹사이트의 검색 및 탐색 기능을 활용할 수 있습니다.

오프라인으로 읽는 것을 선호한다면 가이드를 PDF로 다운로드할 수 있습니다. 오른쪽 탐색 메뉴에 표시되는 "Other Formats > PDF" 옵션을 사용하여 가이드의 개별 섹션을 별도의 PDF로 다운로드할 수도 있습니다.

본문 전체에 걸쳐 차트와 그래프의 예시를 볼 수 있습니다. 이러한 예시 중 상당수에 대해 시각화를 재현할 수 있도록 R 코드를 제공하며, 여러분이 직접 해보고 자신의 목적에 맞게 코드를 수정해 볼 것을 권장합니다.

2.1 이 가이드에 기여하는 방법

제안하거나 질문하기

GitHub 리포지토리의 생각 공유(Discussions) 섹션에서 토론을 시작해 주세요.

버그나 오류를 발견했다면

다음 중 하나를 수행해 주세요:

1. GitHub 리포지토리의 문제(Issues) 섹션에 이슈를 제기합니다. 웹사이트의 'Report an issue' 링크 중 하나를 사용하여 이 작업을 수행할 수 있습니다. 또는
2. 리포지토리를 포크(Fork)하고, 관련 파일을 수정한 후, 우리 리포지토리의 `main` 브랜치에 풀 리퀘스트(Pull Request)를 보냅니다. 웹사이트의 'Edit this page' 링크 중 하나를 사용하여 이 작업을 수행할 수 있습니다.

가이드에 새로운 기능이나 섹션을 추가하고 싶다면

1. GitHub 리포지토리의 문제(Issues) 섹션에 이슈를 제기하고, 'enhancement' 태그를 단 후 제안하는 기여 내용을 설명합니다.
2. 리포지토리를 포크하고, 예를 들어 `my-new-feature-or-section`과 같은 이름으로 새 브랜치를 만듭니다.
3. 브랜치에 콘텐츠, 코드 및 파일을 추가합니다.
4. 우리 리포지토리의 `main` 브랜치에 풀 리퀘스트를 보냅니다.

기여 절차에 대한 자세한 내용은 리포지토리의 README 파일에서 확인할 수 있습니다.

2.2 이 가이드의 다른 사용자들과 교류하는 방법

독자 여러분은 GitHub 리포지토리의 생각 공유(Discussions) 섹션을 최대한 활용하시기 바랍니다. 웹사이트와 리포지토리의 사용 및 모든 기여는 행동 강령의 적용을 받습니다.

2.3 저작권, 재사용, 배포 및 면책 조항

이 가이드의 저작권은 © 2024 Andreas Krause, Nicola Rennie 및 영국 왕립 통계 학회(Royal Statistical Society)에 있습니다. 편집팀이 작성하고 이 가이드에 게시된 콘텐츠는 Creative Commons Attribution 4.0 (CC BY 4.0) International 라이선스 하에 배포됩니다. 이는 원저작자를 표시하는 한, 어떤 목적으로든 사용 및 수정이 가능함을 의미합니다. 이 가이드에 기여함으로써 기여자들은 자신의 작업물을 동일한 조건으로 라이선스하는 것에 동의하게 됩니다.

인용 방법: Krause, Andreas, Nicola Rennie, and Brian Tarran. "Best Practices for Data Visualisation." Royal Statistical Society, 1 February 2024. https://doi.org/10.5281/zenodo.10600718.

이 웹사이트에 게시된 사실 및 의견 진술은 각각의 저자 및 기여자의 것이며 영국 왕립 통계학회(RSS)의 입장과는 다를 수 있습니다.

저자들은 이 웹사이트의 콘텐츠를 책임감 있고 신중하게 준비했습니다. 그러나 저자와 RSS는 이 웹사이트나 다른 링크된 웹사이트 또는 그 이후의 링크에 포함된 정보의 정확성에 대해 명시적이든 묵시적이든 어떠한 보증도 부인합니다. 전체 이용 약관을 참조하십시오.

3 데이터 시각화를 하는 이유

3.1 동기

데이터 시각화는 데이터의 패턴을 식별하고 메시지를 전달하는 매우 효율적인 수단이 될 수 있습니다. 모든 시각화의 과학적 목표는 독자가 데이터를 이해하고 정보를 추출할 수 있도록 하는 것입니다:

- 직관적으로;
- 효율적으로; 그리고
- 정확하게.

시각화를 만들 때는 독자나 의도된 청중의 배경을 고려하는 것이 중요합니다(Krause 2013). 해석은 보는 사람의 눈에 달려 있으며, 시각화는 청중을 염두에 두고 디자인할 때만 메시지 전달에 성공할 수 있습니다.

성공적인 데이터 시각화는 다음과 같은 효과가 있습니다:

- **주의를 끕니다** 수많은 텍스트 속에서 시각화는 눈에 띕니다. 독자가 시간이 부족하거나 문서가 흥미로운지 확신할 수 없는 경우, 시선을 사로잡는 시각화는 독서의 시작을 유도할 수 있습니다.
- **정보 접근성 향상** 텍스트 설명은 길고 읽기 어려울 수 있는 반면, 능숙하게 만든 시각화는 핵심 정보를 더 효율적으로 추출할 수 있게 하여 정보 추출을 즐거운 작업으로 만듭니다.
- **정밀도 증가** 텍스트 설명은 데이터 포인트와 해당 축을 보여주는 시각적 묘사보다 덜 정확한 경우가 많으며, 정확한 숫자가 너무 많은 텍스트는 논의의 흐름을 따라가기 어렵게 만들 수 있습니다.
- **신뢰성 강화** 텍스트 요약은 이야기를 제공하지만, 데이터 시각화는 입증되지 않은 주장에 신뢰성을 더할 수 있습니다. 독자는 숫자를 직접 확인하고 (동일한?) 결론에 도달할 수 있습니다.
- **콘텐츠 요약** 시각적 디스플레이는 복잡한 텍스트 콘텐츠를 요약할 수 있게 하여 독자가 요점을 기억하는 데 도움을 줍니다.

이러한 이유로 데이터 시각화는 과학 논문, 언론 보도, 컨퍼런스 발표, 소셜 미디어 게시물, 비디오 요약 등 거의 모든 종류의 간행물에서 핵심 요소입니다.

표 역시 데이터나 통계를 시각화하는 방법이며, 간행물의 유사하게 중요한 구성 요소가 될 수 있습니다. 어떤 경우에는 표가 그래픽보다 데이터를 더 잘 시각화할 수도 있습니다. 예를 들어, 5개의 숫자는 색상, 각도, 그리고 경우에 따라 음영과 2차원 이상을 사용하는 복잡한 파이 차트보다는 표에 표시하는 것이 더 나을 것입니다.

3.2 데이터 시각화의 간략한 역사

데이터 시각화는 오랫동안 과학 연구 주제이자 다양한 고부가가치 응용 분야를 가진 진화하는 예술 형식이었습니다.

1859년 현대 간호학의 창시자인 플로렌스 나이팅게일(Florence Nightingale)은 러시아와의 전쟁 중 영국군의 위생 상태에 대한 연구 결과를 발표했습니다. 그녀는 표와 차트로 원본 데이터와 요약 통계를 보여주었습니다(Nightingale, 1859). 특히 하나의 차트가 오늘날까지 기념되고 있는데, 바로 "동부 군대의 사망 원인에 대한 다이어그램"이라는 극좌표 영역 차트(polar area chart)입니다.

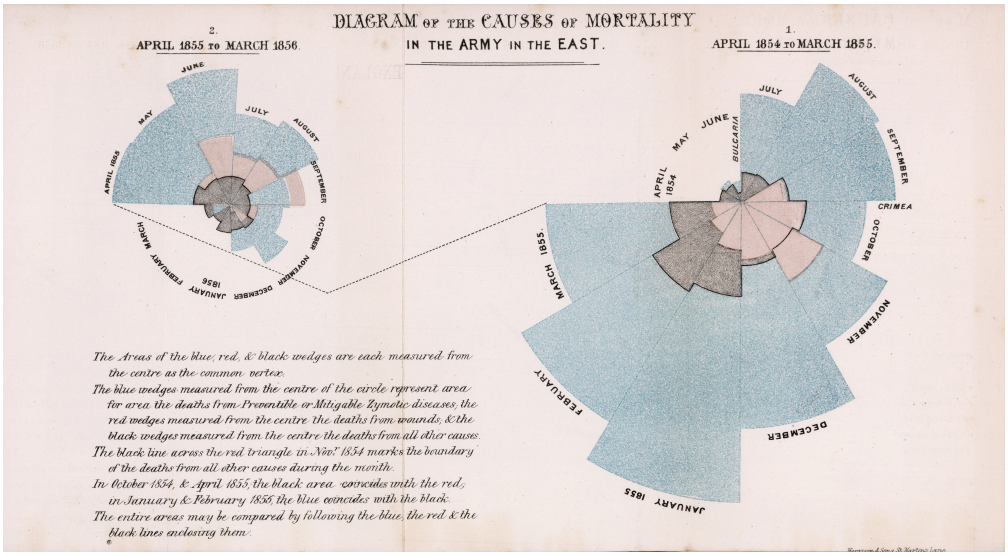


Figure 1: 플로렌스 나이팅게일의 극좌표 영역 차트, "동부 군대의 사망 원인에 대한 다이어그램". 출처: 위키미디어 공용.

나이팅게일의 그래프를 "특히 상징적으로 만든 것은 데이터에 대한 주장을 펼치기 위해 시각적 수사학을 강력하게 사용했다는 점"입니다(Hedley 2020). 이러한 특성은 나이팅게일의 동시대인들이 만든 다른 시각화에서도 분명하게 나타납니다.

콜레라 전파와 관련된 런던의 물 펌프에 대한 단순하지만 꽤 영향력 있는 시각화는 근본 원인 규명의 길을 열었습니다. 의사 존 스노우(John Snow)는 콜레라 사망자에 대한 데이터를 수집하고 해당 런던 주소에 사망자 수를 막대 높이로 나타낸 시각화를 만들었습니다. 이 시각화는 사망자가 브로드 스트리트(Broad Street) 주변에 밀집되어 있음을 보여주었고, 이를 통해 콜레라 전파의 원인인 브로드 스트리트 물 펌프를 식별하는 데 도움을 주었습니다(Snow 1854; Wikipedia contributors 2023).

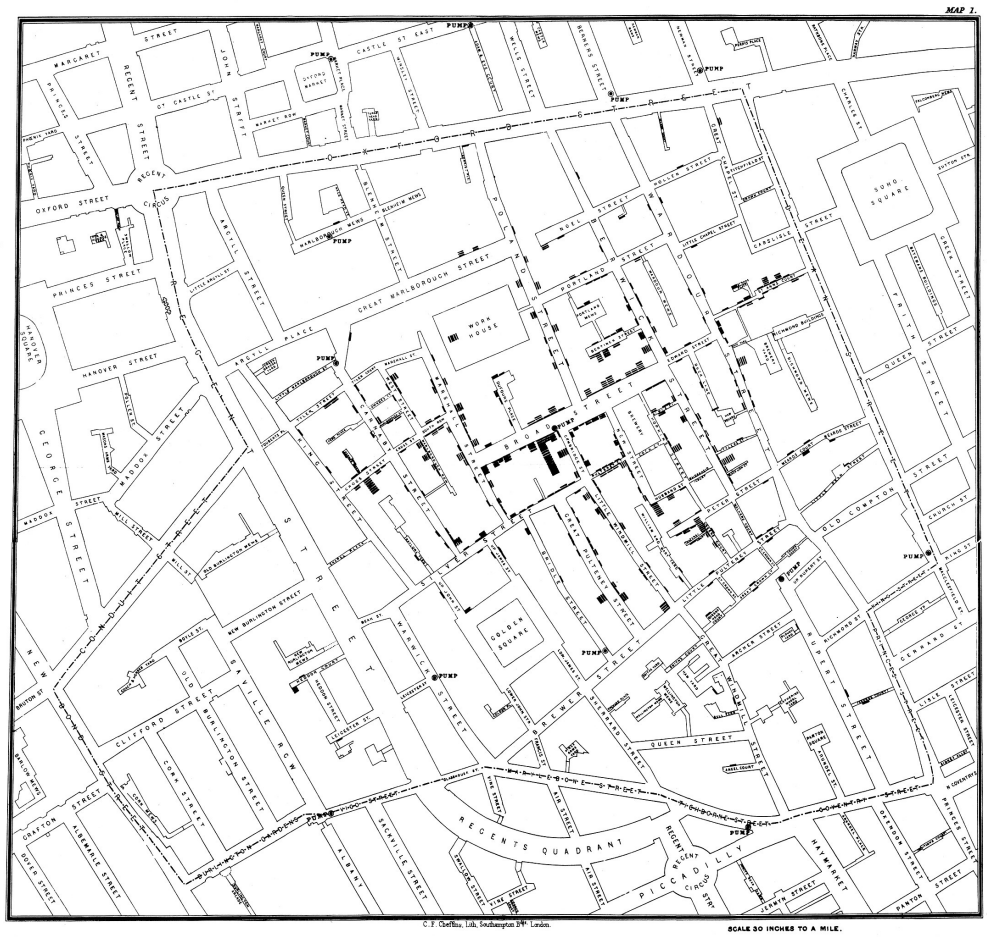


Figure 2: 1854년 런던 전염병 당시 콜레라 사례의 군집을 보여주는 존 스노우의 지도. 출처: 위키미디어 공용.

초기의 복잡한 시각화는 1861년 미나르(Minard)가 1812/13년 나폴레옹의 모스크바 진군과 이후의 퇴각 데이터를 묘사하여 만들었습니다.

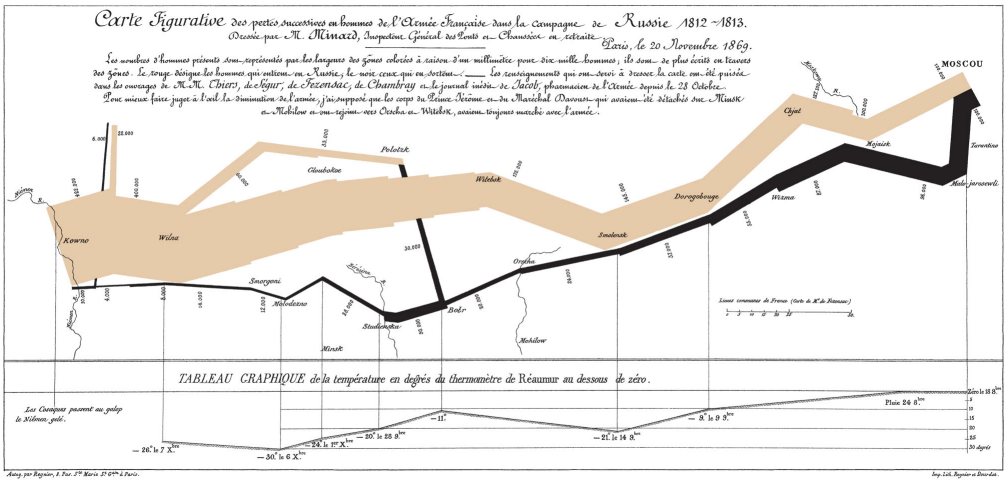


Figure 3: 1869년 샤를 미나르가 그린 "1812-1813년 러시아 원정에서 프랑스군 병력의 연도별 손실" 지도. 출처: 위키미디어 공용.

지도는 군대가 이동함에 따라 위도와 경도를 보여줍니다. 선은 이동 방향을 나타내며 선의 너비는

군대의 규모(생존한 군인 수)를 나타냅니다. 특정 위치는 군대가 있었던 날짜로 표시되었으며 온도도 함께 표시됩니다. 6개의 변수가 단일 디스플레이에 우아하게 엮여 있습니다(Tufte 2001; Corbett 2001; Robinson 1967).

1900년 파리 박람회에서 W. E. B. Du Bois는 흑인 미국인들의 생활상을 보여주는 그래프, 차트, 지도를 전시했습니다(Du Bois 1900; Battle-Baptiste and Rusert 2018).

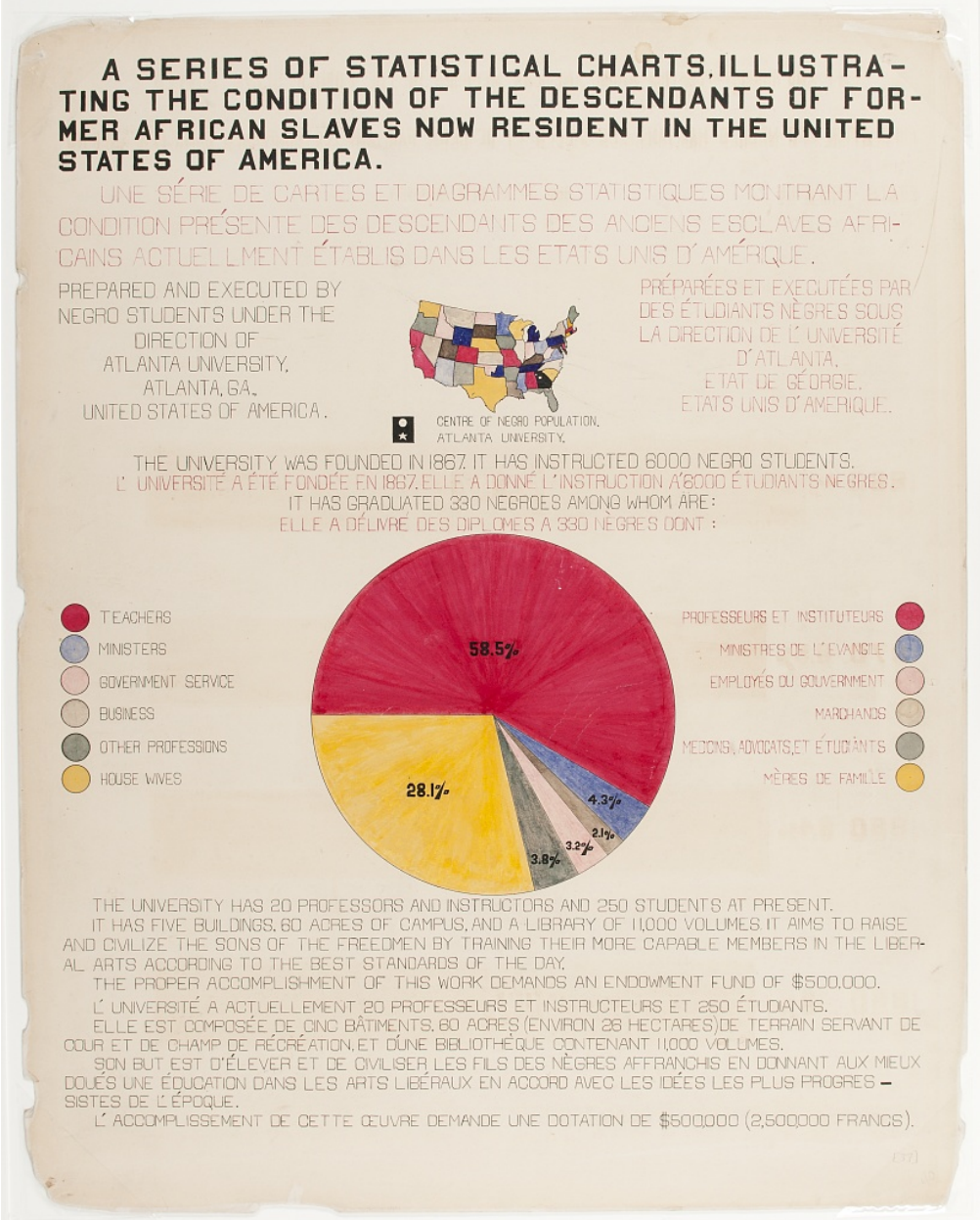


Figure 4: 미국에 거주하는 전 아프리카 노예 후손들의 상태를 보여주는 일련의 통계 차트. 드로잉, ca. 1900. //hdl.loc.gov/loc.pnp/ppmsca.33899. 출처: 미국 의회 도서관.

현대 시각화의 획기적인 작업은 튜키의 책 탐색적 데이터 분석(Exploratory Data Analysis)(Tukey 1977)과 에드워드 터프티(Edward Tufte)(Tufte 1990, 2004, 2006)에 의해 제공되었습니다.

시각화의 역사적 발전은 마이클 프렌들리(Michael Friendly)의 다양한 간행물(Friendly and Denis 2005; Friendly 2018, 2022)에서 제공됩니다.

컴퓨터가 등장하기 전까지 데이터는 손으로 시각화했습니다. 최초의 모니터와 프린터는 텍스트 모드로만 작동했으며 25행 80열 정도의 해상도로 많은 세부 사항이나 정밀도를 허용하지 않았습니다. 그래픽 터미널과 도트 매트릭스 프린터가 뒤를 이었고 레이저 프린터의 개발과 함께 해상도는 계속 증가했습니다.

SAS와 같은 통계 시스템은 일찍부터 데이터 시각화 생성을 가능하게 했습니다("Documentation" n.d.).

아마도 가장 일관된 그래픽 시스템의 구현은 윌킨슨(Wilkinson)의 The Grammar of Graphics(Wilkinson 2005)에 기반한 ggplot(Wickham 2011, 2016a)으로 실현되었을 것입니다.

ggplot2의 핵심 구성 요소는 조건부 디스플레이를 생성하는 기능입니다. 이를 통해 하나 이상의 변수 값으로 데이터를 부분 집합화할 수 있습니다. 이 개념은 클리블랜드(Cleveland)(Cleveland 1993, 1994)가 도입했으며, 널리 알려진 보리 데이터 세트를 선보였습니다. 이 데이터 세트는 6개 농장 중 한 곳에서 두 해 동안의 작물 수확량이 실수로 뒤바뀐 사실이 시각화를 통해 강력하게 시사될 때까지 수십 년 동안 교과서에서 분석되었습니다. 수많은 수치 분석이 놓친 것을 시각화가 밝혀낸 것입니다!

클리블랜드는 자신의 정원에 있는 격자 구조물(trellis)에서 영감을 받아 디스플레이 유형을 "trellis"라고 명명했습니다(Cleveland 1993). Trellis 개념은 S(Becker and Chambers 1984)와 S-PLUS(Becker and Cleveland 1996)에서 처음 구현되었습니다. 1980년대 뉴질랜드 오클랜드에서 로버트 젠틀맨(Robert Gentleman)과 로스 이하카(Ross Ihaka)가 R(R Core Team 2021)을 개발했을 때, "trellis"가 상표(Lucent와 이후 MathSoft)로 등록되어 있었기 때문에 `lattice`라는 패키지가 개발되었습니다(Sarkar 2008). ggplot의 등장으로 패싯(faceting, facet_grid 및 facet_wrap 함수 사용)이라는 용어가 "lattice"와 그 "패널(panels)"을 대체했습니다.

4 시각화의 원칙과 요소

데이터 시각화는 목적을 충족해야 합니다. 시각화의 목적을 이해함으로써 우리(저자 또는 독자로서)는 시각화가 목표를 달성했는지 아니면 개선이 필요한지를 평가할 수 있는 위치에 서게 됩니다!

아마도 시각화의 가장 일반적인 목적은 서로 다른 치료를 받는 환자 데이터와 같이 데이터 그룹을 비교하는 것입니다. 축, 축 제한, 레이블 및 기호를 잘 선택하면 데이터의 패턴을 식별하는 데 상당히 도움이 되지만, 이러한 요소 중 하나라도 잘못 선택하면 정보 추출을 상당히 방해할 수 있습니다.

4.1 차트의 요소

시각화의 다양한 요소는 정보를 추출하는 효율성에 기여할 수 있습니다. 모든 시각화 소프트웨어 패키지는 기본 스타일로 그래픽을 출력하지만, 이것이 여러분이 만들고 있는 데이터 시각화에 대한 최적의 선택인 경우는 드뭅니다. 데이터 시각화를 디자인할 때는 사용 가능한 다양한 옵션을 고려해야 합니다. 이러한 옵션 중 일부는 다음과 같습니다.

레이아웃 (패널, 패싯)

레이아웃(여러 패널, 패싯 또는 서브플롯의 배열)은 효율적인 비교와 매우 관련이 있습니다. y축 데이터를 비교해야 하는 경우 모든 패널을 수평으로 정렬하여 단일 y축을 사용하는 것이 비교를 용이하게 하는 반면, x축 값을 효율적으로 비교하려면 패널을 쌓아야 합니다. 매트릭스 레이아웃(단일 그림에 여러 행과 열)은 개별 패널에 표시된 데이터가 관련이 없거나 공간이 단일 행 또는 단일 열을 허용하지 않는 경우(예: 패널이 너무 많아 한 행에 맞지 않는 경우)에만 사용해야 합니다.

예: 아래 그림은 레이아웃을 제외하고는 정확히 동일한 데이터와 정확히 동일한 유형의 시각화를 보여주는 두 가지 다른 레이아웃을 보여줍니다. 왼쪽 패널의 데이터를 비교하는 것이 얼마나 어려운지, 반면 오른쪽 레이아웃에서는 얼마나 쉬운지 주목하십시오. 핵심 차이점은 오른쪽 패널이 공통 y축을 공유한다는 것이며, 이는 패널 간의 y값 비교에 핵심적입니다.

```
# ---
# Data set creation.

set.seed(93384)

time <- c(0, 0.5, 1, 2, 4, 8, 12, 16, 24)
n <- 32 # no of subjects

data <- expand.grid(ID = 1:n, time = time)

bw <- data.frame(
  ID = sort(unique(data$ID)),
  bw = rlnorm(n, log(75), sdlog = 0.25)
)

bw$bw.category <- cut(bw$bw,
  breaks = quantile(bw$bw, c(0, 0.33, 0.66, 1)),
  labels = paste(c("low", "medium", "high"), "body weight"),
  include.lowest = TRUE
)

data <- merge(data, bw)

data <- data[order(data$ID, data$time), ]

# Simulate drug concentrations as a function of body weight.
data$conc <- 100 / (data$bw^1.0) * exp(-0.085 * data$time) *
  rlnorm(nrow(data), sdlog = 0.25) + # res. error
  (data$ID - mean(data$ID)) / mean(data$ID) / 4 # r. eff

# ---
# Visualisation.
library(ggplot2)

gg <- list()

data$ID <- factor(data$ID)

gg[["3x1"]] <- ggplot(data, aes(x = time, y = conc, group = ID, color = ID)) +
  geom_line()
gg[["3x1"]] <- gg[["3x1"]] + scale_x_continuous(breaks = seq(0, 24, by = 4))
gg[["3x1"]] <- gg[["3x1"]] + theme_bw() +
  xlab("time [h]") +
  ylab("drug concentration [ng/mL]")
gg[["3x1"]] <- gg[["3x1"]] + facet_grid(bw.category ~ .)
gg[["3x1"]] <- gg[["3x1"]] + theme(legend.position = "none")
```

```
gg[["1x3"]] <- gg[["3x1"]] + facet_grid(. ~ bw.category)

# Add space to the rhs of the first figure for better separation in the cowplot.
gg[["3x1"]] <- gg[["3x1"]] +
  theme(plot.margin = unit(c(0.5, 4, 0.5, 0.5), "lines"))

# Both figures into a single output figure.
library(cowplot)
plot_grid(gg[[1]], gg[[2]], rel_widths = c(1.5, 2))
```

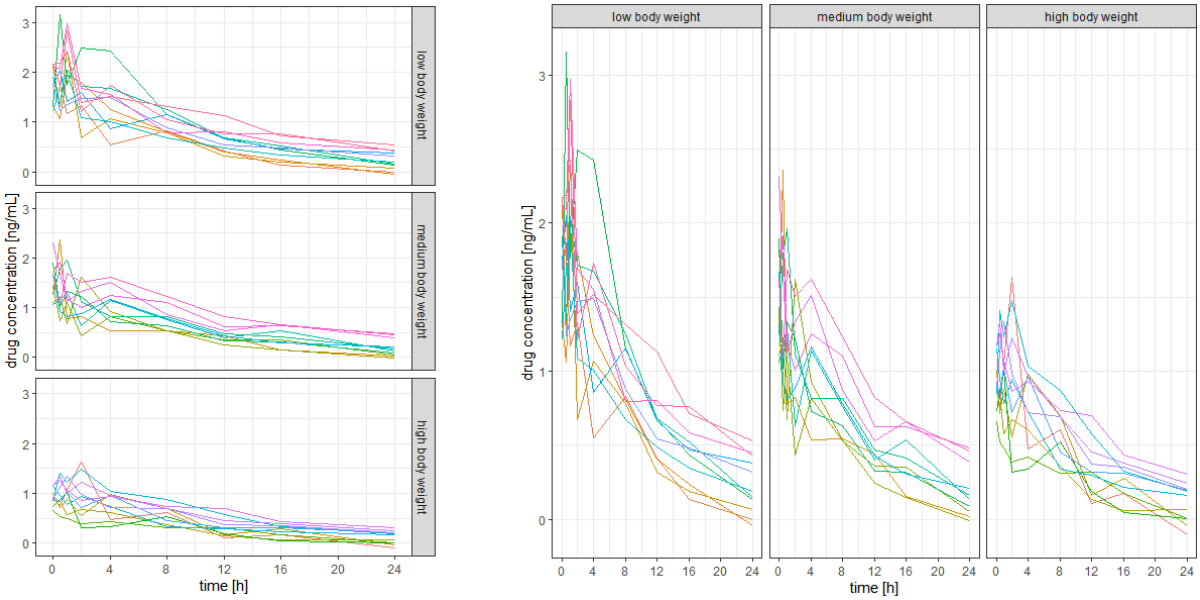


Figure 5: 레이아웃 선택이 패널 간 비교 용이성을 결정하는 방식을 보여주는 선 도표. 왼쪽 패널은 세로로 쌓인 3개의 플롯을 보여줍니다. 오른쪽 패널은 가로로 배열된 동일한 3개의 플롯을 보여줍니다.

종횡비 (Aspect ratio)

디스플레이에서 데이터에 대한 시각적 인식은 x축과 y축의 선택에 의존해서는 안 됩니다. 많은 경우 정사각형 그림은 시각적 편향을 방지합니다. 특히 두 축이 어떤 사건 전후의 측정, 관측 데이터 대 모델 예측 값, 또는 더 일반적으로 동일한 단위(kg 또는 m 등)와 같은 공통점을 공유하는 경우 정사각형 그림을 고려해야 합니다. 일반적으로 1:1 종횡비(x 방향과 y 방향에서 1 측정 단위의 물리적 길이)가 좋은 기본값입니다. 그림이 x축과 y축에 대해 동일한 범위를 갖는 경우 1:1 종횡비는 정사각형 그림을 산출합니다.

예: 아래 세 패널은 모두 동일한 데이터를 보여주며, 모델 적합도(예측 값 대 관측 값)에 대한 판단을 가능하게 하는 것을 목표로 합니다. 늘어난 x축, 늘어난 y축, 그리고 동일한 축 제한을 가진 1:1 종횡비 간의 시각적 인식 차이에 주목하십시오.

```
# Observed vs predicted (any data with comparable x and y will do).

# ---
# Data set.
# Old Faithful Geyser (Yellowstone) data set with eruption duration
```

```r
#   and waiting time to the next eruption (both in minutes).
data <- data.frame(
  x = faithful$eruptions,
  y = faithful$waiting
)

# ---
# Regression model fit.
fit <- lm(y ~ x, data = data)

# Addition of predicted values to the data set.
data$pred <- predict(fit)

# Range of y and y predicted combined.
r <- range(unlist(data[c("y", "pred")]))

# ---
# Plotting.

library(ggplot2)

gg <- ggplot(data, aes(x = pred, y = y))

# Adding the line of identity, y = x
# (note: plotting it first will add points on top).
gg <- gg + geom_abline(intercept = 0, slope = 1, color = "black", linewidth = 1)

# Adding points, removing grey background.
gg <- gg + geom_point() + theme_bw()

# Adding regression fit (local smoother, loess) of y~x.
gg <- gg + geom_smooth(method = "loess", color = "firebrick", se = FALSE)

# Adding axis labels.
gg <- gg + xlab("predicted") + ylab("observed")

# Aspect ratios are commonly not fixed but adapted to figure size.
#   With dynamic displays, the point of different perception
#   might not be obvious depending on the figure/screen size.
#   To make that point independent of figure height and width,
#     the aspect ratio is fixed in this example.
gg <- gg + coord_fixed(ratio=0.5)

# Copy the figure and fix the aspect ratio to 2, i.e.,
#   one pixel in x corresponds to 2 pixels in y.
gg2 <- gg + coord_fixed(ratio=2)

# Setting the aspect ratio to 1 (1 unit in x and y
#   corresponds to the same number of pixels) and
#   setting axis limits to be identical.
gg3 <- gg + coord_fixed(ratio=1, xlim=r, ylim=r)
```

```
# Cow (column-wise) plot, combine all figures into one.
library(cowplot)
plot_grid(gg, gg2, gg3, rel_widths = c(4, 2, 3), nrow = 1)
```

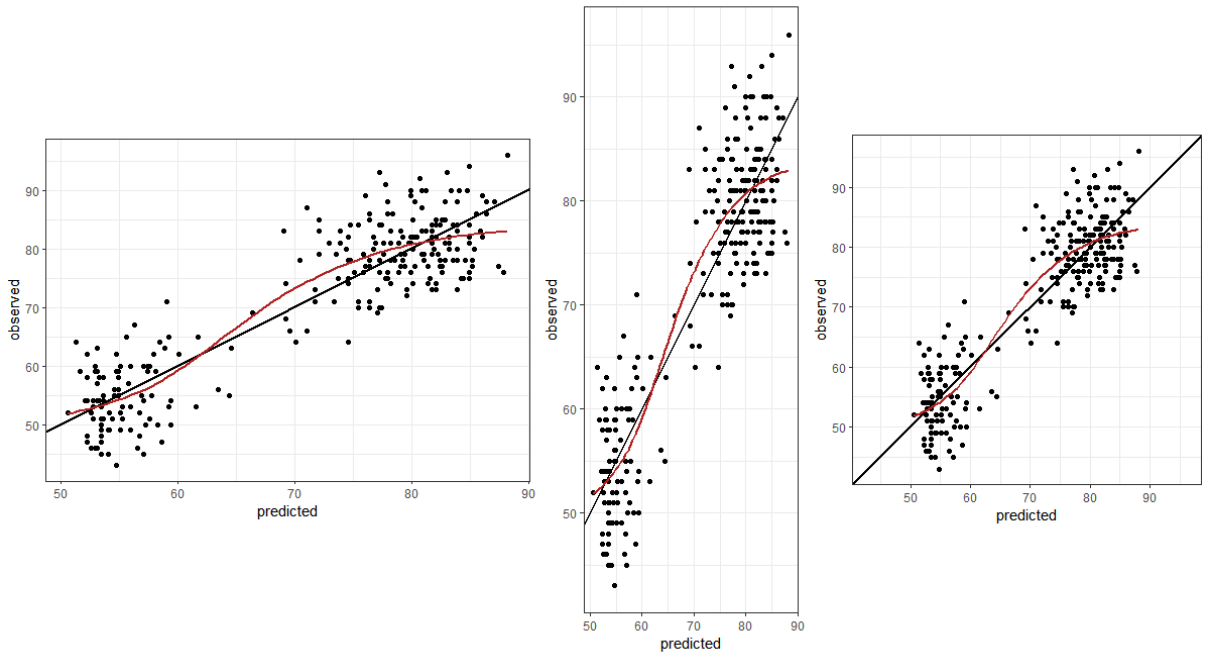


Figure 6: 모델 적합도에 대한 시각적 인식이 종횡비에 의해 어떻게 영향을 받을 수 있는지 보여주는 산점도.

선 (Lines)

선은 순서, 즉 시퀀스를 도입합니다. 순서가 없는 경우 데이터는 선으로 서로 연결되어서는 안 됩니다. 다른 선 유형이 고유한 순서를 가진 다른 그룹을 나타내는 경우, 선택한 선 스타일도 순서를 갖는 것이 좋습니다(예: 가장 낮은 그룹에서 가장 높은 그룹으로 갈수록 선 두께, 대시 밀도 또는 색상의 어두움이 증가함).

점 (Points)

점 기호는 시각화가 데이터를 의미 있는 방식으로 표시하도록 선택해야 합니다. 수천 개의 데이터 포인트를 표시해야 하는 경우, 겹치는 채워진 기호보다 열린 원이나 더 작은 채워진 기호가 더 좋습니다. 데이터가 이산적이고 관측치가 겹치는 경우 더 나은 시각화를 위해 데이터를 부드럽게 지터링(jittering)하는 것을 고려하십시오.

색상 (Colours)

색상은 서로 다른 그룹을 식별하는 데 도움을 주는 것과 같은 목적을 수행해야 합니다. 순수 장식용 색상은 권장하지 않습니다. Tufte는 “색상은 자연스러운 시각적 계층 구조를 가지고 있기 때문에 다양한 회색 음영이 색상보다 다양한 양을 더 잘 보여준다”며 “회색 음영은 데이터 측정에 쉽게 이해되는 순서를 제공한다. 이것이 핵심이다”라고 지적했습니다 (Tufte 2001, 154).

축 (Axes)

원점 및 제한

다른 범위를 선택할 타당한 이유가 없는 한 축 범위는 일반적으로 0에서 시작해야 합니다. 데이터에 음수 값이 포함되지 않은 경우 축은 음수 값으로 뻗어 나가서는 안 되며 특히 음수 값에 눈금을 표시해서는 안 됩니다.

상대적인 변화나 비율이 표시되는 경우 해당 축은 로그여야 하며(예: 1/4이 기준점 1까지의 거리가 4와 같도록), 변화 없음 지점을 중심으로 대칭이어야 합니다. 변화 없음 지점에 보조선을 그으면 해석이 쉬워집니다. 축 눈금과 보조 회색 선은 값을 읽는 것을 용이하게 하여 보는 사람이 잘못된 선형 보간을 하는 것을 방지합니다. 축 눈금 레이블은 비율을 나타내야 합니다(예: 0.25 대신 "1/4").

x와 y로 표시되는 데이터가 비교 가능한 경우 축 제한은 동일해야 하며 그림은 정사각형이어야 하므로 x 방향과 y 방향의 거리가 일관되어야 합니다.

예: 아래 그림은 의도적인 축 제한 선택(여기서는 y축 제한)이 인식에 큰 차이를 만들 수 있으며, 따라서 독자의 해석에도 큰 차이를 만들 수 있음을 보여줍니다. 두 패널의 유일한 차이점은 y축 범위입니다.

```r
library(ggplot2)
plot_data <- data.frame(
  type = factor(
    c("Our product", "Competitor"),
    levels = c("Our product", "Competitor")
  ),
  value = c(220, 210)
)

# Original plot
ggplot(plot_data) +
  geom_col(
    mapping = aes(x = type, y = value),
    fill = "lightblue",
    colour = "black"
  ) +
  scale_y_continuous(breaks = seq(0, 220, by = 20), expand = c(0, 0)) +
  labs(x = "", y = "") +
  theme_minimal()

# Offset the y axis
offset <- 208
ggplot(plot_data) +
  geom_col(
    mapping = aes(x = type, y = value - offset),
    fill = "lightblue",
    colour = "black"
  ) +
  scale_y_continuous(
    breaks = seq(0, 14, by = 2),
    labels = seq(0 + offset, 14 + offset, by = 2),
    expand = c(0, 0)
  ) +
```

```
  labs(x = "", y = "") +
  theme_minimal()
```

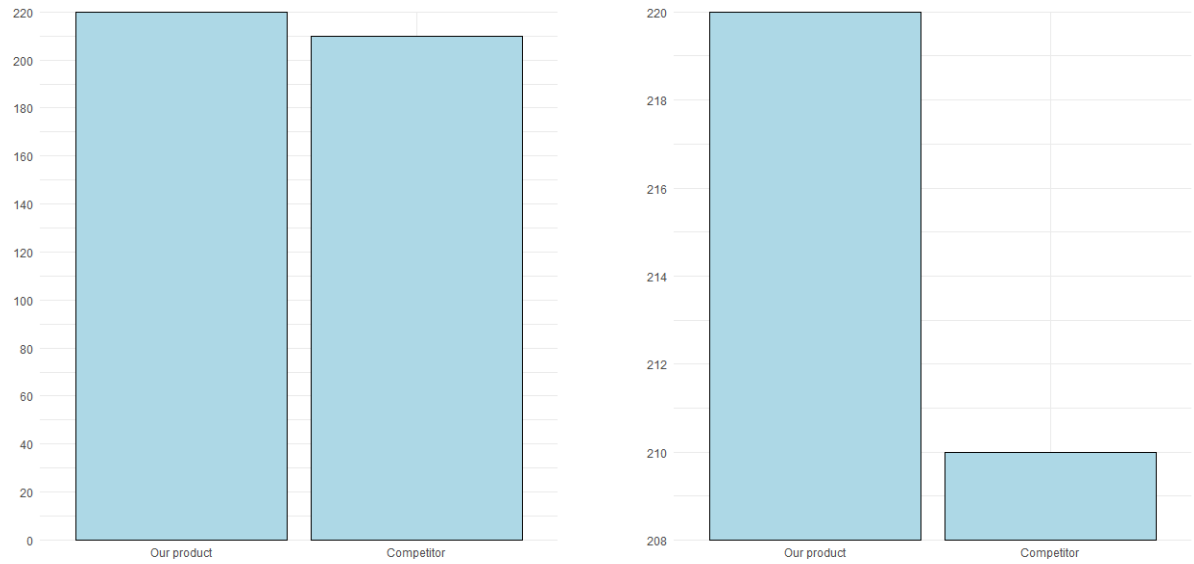


Figure 7: 축 제한의 선택에 의해 시각적 인상이 어떻게 좌우될 수 있는지 보여주는 막대 차트.

선형 및 로그 축

예: 어떤 변수 y가 다른 변수 x에 따라(참조 또는 기준 측정과 비교하여) 변하는지 알고 싶다고 가정해 봅시다. 예를 들어, 혈압이 치료에 따라 변하는가?

아래 그림은 참조(기준) 측정과 비교하여 평균적으로 변화가 없는 데이터(빨간색 가로선, 상단 두 패널)를 보여줍니다. 데이터의 약 절반은 변화 없음 지점 아래에 있고 나머지 절반은 위에 있습니다. 그러나 선형 축을 사용하는 왼쪽 상단 패널에서는 이것이 쉽게 드러나지 않으며 공간은 참조선에서 멀어지는 수직 방향(위아래 모두)으로 비대칭적으로 할당됩니다. 로그 축을 사용하는 오른쪽 상단 패널에서는 참조선 주위에 데이터 포인트가 비교적 고르게 퍼져 있는 것을 명확하게 볼 수 있습니다. 아래쪽 행에서는 동일한 데이터가 다시 선형 및 로그 축을 사용하여 히스토그램 형식으로 표시됩니다.

```
# Naïve plot of y vs x. If there is no change (on average),
# half the data are below the line of no change.
# Asymmetric view, and it depends on y/x or x/y.

set.seed(33838)
x <- data.frame(
  x = rlnorm(200, 2, 0.2),
  y = rlnorm(200, 0.2, 0.75)
)
# Add an outlier manually.
x <- rbind(x, data.frame(x = quantile(x$x, 0.8), y = max(x$y) * 1.5))

# ---
# Plotting.

library(ggplot2)
```

```r
# Scatterplot of y vs x.
gg <- ggplot(x, aes(x = x, y = y)) +
  geom_point() +
  theme_bw()
gg <- gg + geom_hline(yintercept = 1, color = "firebrick", linewidth = 2)
gg <- gg + xlab("x-variable") + ylab("Fold-change")
gg
# Logarithmic axes, symmetric range (!):
xbr <- c(1 / 10, 1 / 5, 1 / 2, 1, 2, 5, 10)
gg <- gg + scale_y_continuous(
  breaks = xbr, trans = "log10",
  limits = max(abs(x$y))^c(-1, 1)
)
gg

# Second axis:
gg <- gg + scale_y_continuous(
  breaks = xbr,
  labels = paste(100 * xbr, "%", sep = ""),
  trans = "log10",
  limits = max(abs(x$y))^c(-1, 1),
  sec.axis = sec_axis(
    trans = ~ . * 1, breaks = xbr,
    labels = ifelse(xbr < 1, paste("1/", 1 / xbr, sep = ""), xbr)
  )
)

# ---
# Univariate distribution (histogram).

gg <- ggplot(x, aes(x = y)) +
  theme_bw() +
  xlab("Fold-change")
gg <- gg + geom_histogram(color = "firebrick", fill = "gray")
gg

# Symmetric range, log scale.
gg <- gg + scale_x_continuous(
  breaks = xbr,
  labels = ifelse(xbr < 1, paste("1/", 1 / xbr), xbr),
  trans = "log10",
  limits = max(abs(x$x))^c(-1, 1)
)
gg
```

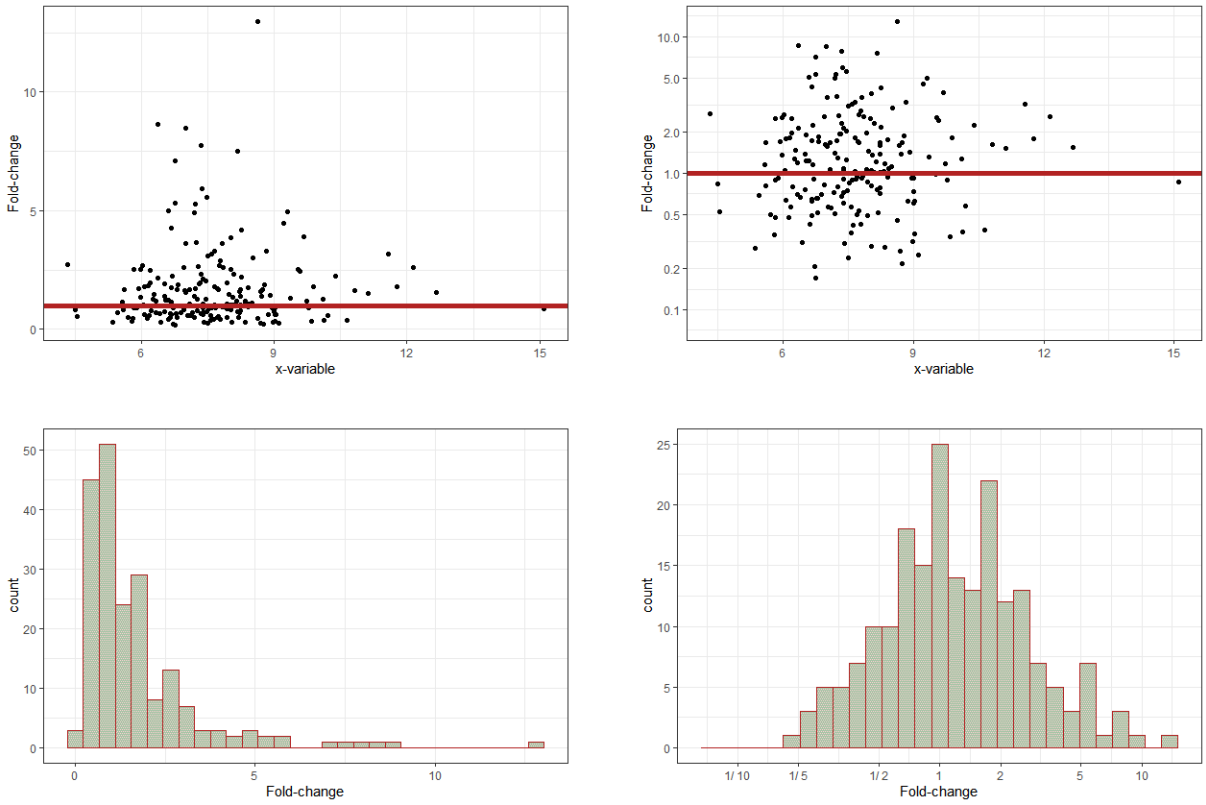


Figure 8: 선형 축(왼쪽)에서 로그 축(오른쪽)으로 전환하면 일부 기준 측정에 대한 값의 변화를 명확히 하는 데 도움이 됩니다.

기호 (Symbols)

기호는 직관적이어야 합니다(예: 긍정적 결과는 “+”, 부정적 결과는 “–”, 중립적 결과는 “O”). 이상적인 경우, 기호는 범례를 한 번만 봐도 이후 시각화에 집중할 수 있어야 합니다.

삼각형, 원, 사각형과 같은 기호는 직관적인 함축적 의미를 갖지 않습니다. 그러나 데이터에 순서가 있는 경우 기호의 순서(예: 꼭짓점 수(원, 대시, 삼각형, 사각형, 오각형 등))에 반영될 수 있습니다.

범례 (Legends)

범례는 너무 많은 주의를 끌어서는 안 되며 데이터를 가려서는 절대 안 됩니다. 그림의 여백에 배치하거나 그림 아래에 작은 텍스트로 캡처할 수 있습니다.

범례 항목이 단일 객체(예: 그룹당 한 줄)를 참조하는 경우 범례를 해당 객체 옆에 배치하면 독자가 범례 항목을 디스플레이 객체에 매핑하기가 더 쉬워집니다.

예: 아래 그림은 해당 데이터 옆에 그림에 직접 범례를 추가하여 가독성을 높일 수 있음을 보여줍니다. 레이블은 겹치지 않아야 하며 레이블 위치는 데이터에 따라 조정이 필요할 수 있습니다.

```
# ---
# EU stock markets, year and indices DAX, SMI, CAC, and FTSE.

# Store graphics into a list.
gg.list <- list()
```

```r
# Prepare the data set (reformat EuStockMarkets that comes with R).
x <- EuStockMarkets
df <- data.frame(
  time = as.vector(time(x)),
  index = rep(colnames(x), rep(nrow(x), ncol(x))),
  value = as.vector(x),
  stringsAsFactors = TRUE
)
df$index2 <- df$index # For use with labels later.

library(ggplot2)

# Standard layout and legend.
gg <- ggplot(df, aes(x = time, y = value, group = index, color = index, label = index2))
gg <- gg + geom_line() + theme_bw()

# Nicer axis tick mark settings.
ax <- pretty(df$time, n = 10)
gg <- gg + scale_x_continuous(limits = range(ax), breaks = ax)
gg <- gg + xlab("year") + ylab("Stock index")

gg.list[[1]] <- gg

# Use the last element of each time series for x,y of the label.
# Use that the last element is the first element of the reversed order,
#   and extract the first element per index by using !duplicated.
y <- df[rev(order(df$time)), ] # descending in time.
y <- y[!duplicated(y$index), ] # first entry per index
y$index2 <- y$index # Create a copy that contains formatted strings.
levels(y$index2)[levels(y$index2) == "FTSE"] <- "\n\nFTSE"
# Add a newline to separate FTSE from DAX.
# Note that the factor level is modified, not the data.

# Drop the legend, move labels into figure.
gg <- gg + geom_text(data = y, hjust = "left", nudge_x = 0.1)
# aes as before, nudge adds space on the lhs.
gg <- gg + theme(legend.position = "none")
gg.list[[2]] <- gg

# ---
# Both figures into a single output figure.

library(cowplot)
plot_grid(gg.list[[1]], gg.list[[2]], rel_widths = c(2.25, 2))
```

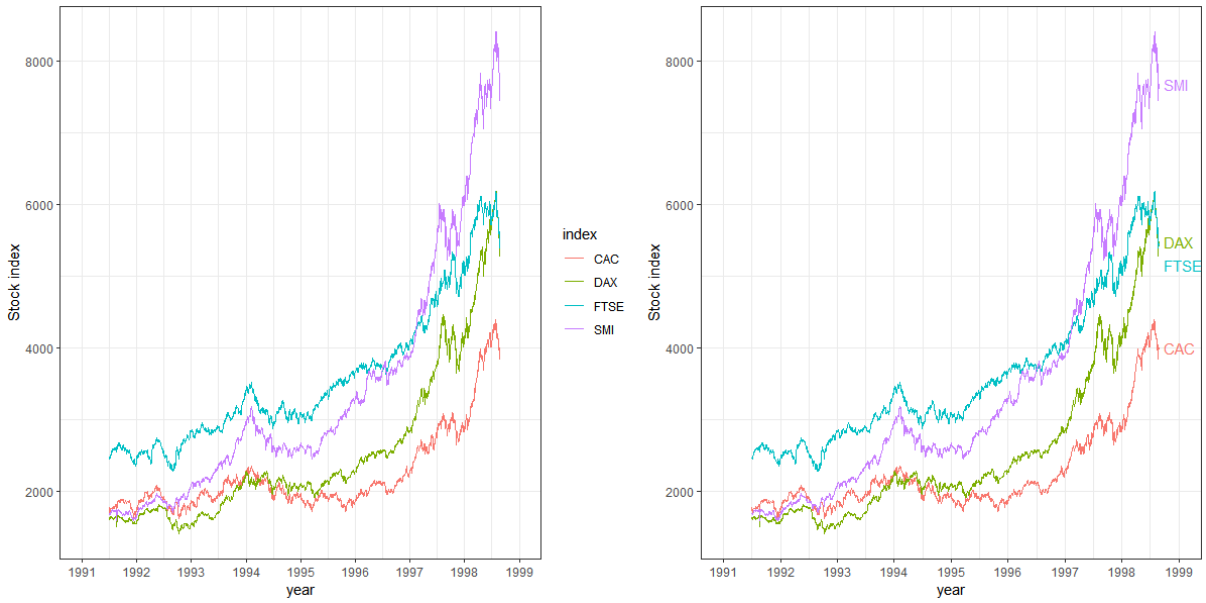


Figure 9: 그림 요소에 직접 레이블을 지정하면 가독성이 어떻게 향상될 수 있는지 보여주는 선 도표.

방향 (Orientation)

예를 들어 막대 그래프로 시각화해야 할 순서가 데이터에 있는 경우, 데이터를 가로 막대로 표시하고 위에서 아래로(가장 높은 값에서 가장 낮은 값으로) 정렬하는 것이 세로 막대를 왼쪽에서 오른쪽 방향으로 표시하는 것보다 더 직관적입니다(Few 2004, 182). 이에 대한 일반적인 예외는 데이터를 시간 단위에 따라 정렬해야 하는 경우입니다. 시간은 종종 과거에서 현재, 미래로, 왼쪽에서 오른쪽으로 흐르는 것으로 시각적으로 해석됩니다.

예를 들어 상자 그림의 경우 가로 방향이 세로 방향보다 더 정확한 시각적 비교가 가능합니다. 아래 그림에서 볼 수 있듯이 인간의 눈은 가상의 수평선보다 가상의 수직선을 따라가기가 더 쉽기 때문입니다. 또한 왼쪽 패널에서는 긴 x축 레이블이 겹쳐서 읽을 수 없지만 오른쪽 패널에서는 가로 방향으로 읽을 수 있습니다.

```
# Store figures into a list.
gg.list <- list()

library(ggplot2)

x <- mpg # miles per gallon data set.
x$car <- paste(x$manufacturer, x$model)

gg <- ggplot(x, aes(x = car, y = hwy, group = car))
gg <- gg + geom_boxplot() + theme_bw() + xlab("Miles per gallon (highway)")
gg.list[["vertical orientation"]] <- gg

gg.list[["horizontal orientation"]] <- gg + coord_flip()

# ---
# Both figures into a single output figure.
```

```
library(cowplot)
plot_grid(gg.list[[1]], gg.list[[2]], rel_widths = c(2, 2.5))
```

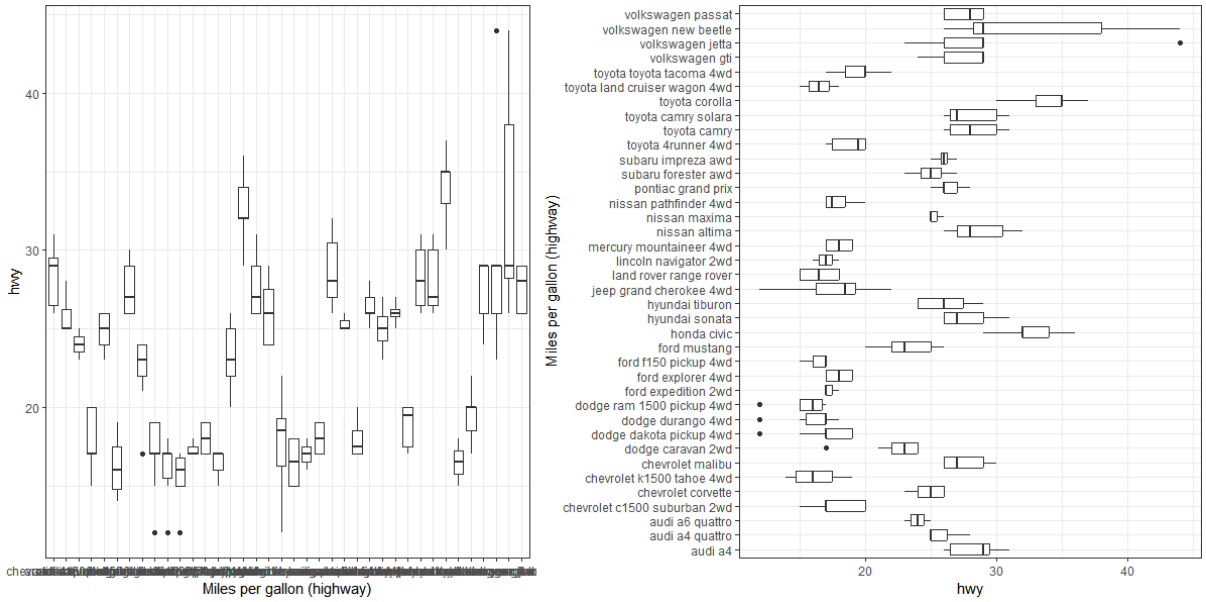


Figure 10: 방향을 변경하면 레이블의 가독성과 데이터의 시각적 해석을 모두 개선할 수 있습니다.

보조 요소 (Auxiliary elements)

일반적으로 데이터 시각화를 디자인할 때 우리는 Tufte가 "차트 정크(chart junk)"라고 부르는 것, 즉 디스플레이의 정보를 추가하거나 향상시키지 않는 차트의 모든 요소를 경계해야 합니다. 물론 데이터가 쇼의 주인공이므로 가장 많은 공간을 차지해야 합니다. 한편, 모든 추가 요소(선, 색상, 기호 등)는 데이터 해석에 대한 기여도를 평가해야 합니다. 기여도가 없다면 제거할 수 있습니다. 데이터 시각화에서 불필요한 요소의 전형적인 예는 각각 x축과 y축이 있는 5개의 패널(하위 그림 또는 패싯)이 있는 그림입니다. 다른 패널에 표시된 데이터의 효율적인 비교를 허용하기 위해 패널이 모두 한 행에 정렬된 경우 단일 y축이면 충분합니다. 이렇게 하면 데이터를 위한 더 많은 공간이 생기고 부가적인 이점으로 y축 제한과 데이터 위치가 동기화됩니다.

도움이 되는 추가 요소에는 관련 지점의 선이 포함될 수 있습니다. 예를 들어 x=0(예: 시간 0의 경우)의 수직선, y=0(예: 변화 없음 지점을 나타내기 위해)의 수평선 또는 대각선(y=x, 예: x와 y 사이에 차이가 없음을 나타내기 위해)이 있습니다. 보조 요소를 데이터 아래에 그릴지 아니면 위에 그릴지 고려하십시오(일반적으로 그래픽 요소 추가 순서에 따라 다름).

로컬 산점도 평활화(loess, lowess, polynomial)는 제한된 모델 가정으로 관계를 식별하는 데 특히 도움이 될 수 있습니다. 신뢰 대역(점별 신뢰 구간)은 관련된 경우에만 표시해야 합니다.

예: 아래 그림은 시작점(시간 0)에서 시간이 지남에 따른 변화를 보여줍니다. 일반 그림(왼쪽 패널)은 독자가 y축 레이블을 읽어 변화 없음 지점(y=0)을 식별해야 합니다. 변화 없음 지점을 나타내는 선을 추가(가운데 패널)하면 읽기가 더 쉬워지고, 변화 없음 지점을 중심으로 대칭인 축 범위를 선택(오른쪽 패널)하면 증가(위쪽 변경)는 시각화의 위쪽 절반에, 감소(아래쪽 변경)는 아래쪽 절반에 할당되어 읽기와 해석이 더 직관적이 됩니다.

```
# Function for data set generation.
make.data <- function(
    x = c(0, 0.5, 1, 2, 4, 8, 12, 16, 24),
```

```r
    y = exp(-0.2 * x) - exp(-0.21 * x),
    sd = 0.25, # std dev of y
    seed = 4384590,
    n = 50) {
  # Setting the random number seed for reproducibility.
  set.seed(seed)
  # Creation of x- and y-variables.
  x2 <- rep(x, n)
  y2 <- NULL
  for (i in 1:n) {
    y2 <- c(y2, y * (2 * (n / 4 - i)) + rlnorm(length(y), sd = sd))
  }

  # Creation of an identifier for each profile.
  ID <- factor(rep(1:n, rep(length(y), n)))

  # Composition of the data set.
  df <- data.frame(PD = 100 * y2, time = x, ID = ID)

  # Addition of a baseline variable.
  BL <- df[df$time == 0, c("ID", "PD")]
  names(BL) <- c("ID", "BL")
  df <- merge(df, BL)

  # Addition of change from baseline.
  df$Change <- df$PD - df$BL

  # Definition of treatment.
  df$trt <- ifelse(df$BL > mean(df$BL), "active", " placebo")

  return(df)
}

# Generate the data.
x <- make.data()

# ---
# Figures.

library(ggplot2)
gg <- ggplot(x, aes(x = time, y = Change, group = ID, color = ID))
gg <- gg + theme_bw()
gg <- gg + xlab("Time [h]") + ylab("Change from baseline")
gg <- gg + geom_line(linewidth = 1.1) + theme(legend.position = "none")
gg <- gg + facet_grid(. ~ trt)

# Addition of an auxiliary line at y=0.
gg2 <- gg + geom_hline(yintercept = 0, linewidth = 1.2)

# Symmetric y-axis limits.
gg3 <- gg2 + ylim(c(-1, 1) * max(abs(x$Change)))
```

```
# Arranging all plots into one figure.
library(cowplot)
plot_grid(gg, gg2, gg3, nrow = 1)
```

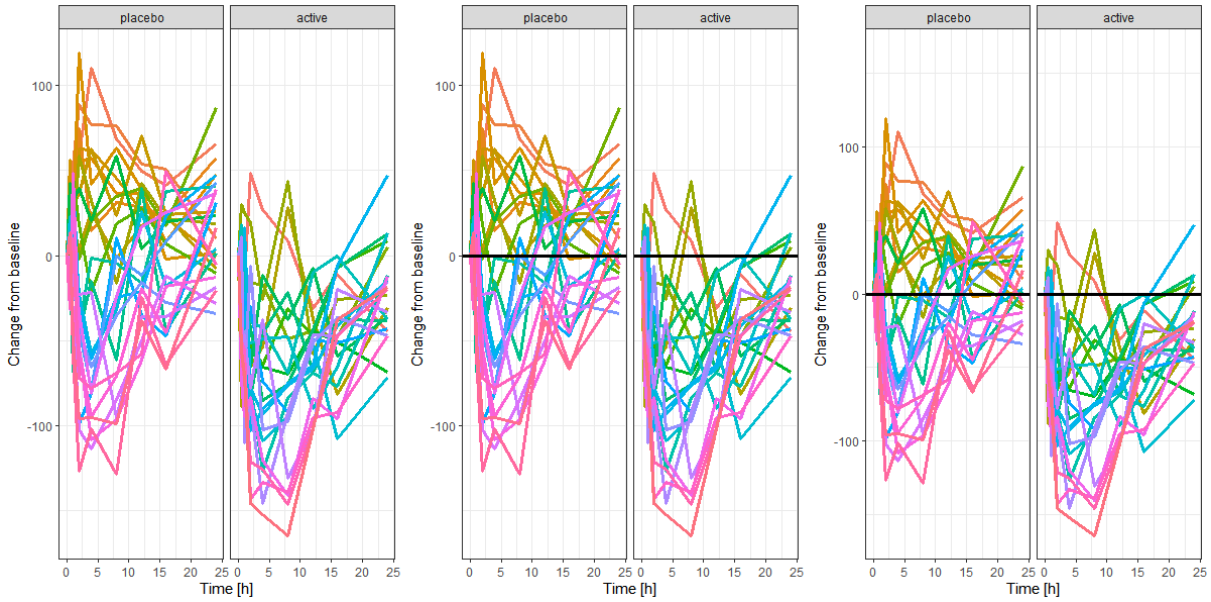


Figure 11: 변화 없음 지점을 나타내는 선과 같은 요소를 추가하면 가독성을 향상시킬 수 있습니다.

3차원 차트

3차원 데이터 디스플레이는 몇 가지 면에서 눈에 띌 수 있지만, 이러한 프레젠테이션으로 정확한 읽기와 해석은 쉽지 않습니다.

예: 아래 그림의 왼쪽 패널은 단일 숫자 10을 3D 막대 차트로 보여줍니다. 값의 올바른 식별은 빨간색 "X"로 표시됩니다. 막대의 높이는 보는 위치에서 뒤쪽 벽(축)에 투영되어야 합니다. 오른쪽 패널은 기울어진 표면에 막대를 배치하여 높이 차이가 없을 때 높이 차이가 있는 것처럼 인식되도록 합니다. 각 행과 열에 동일한 4개의 숫자 10, 20, 30, 40이 표시됩니다.

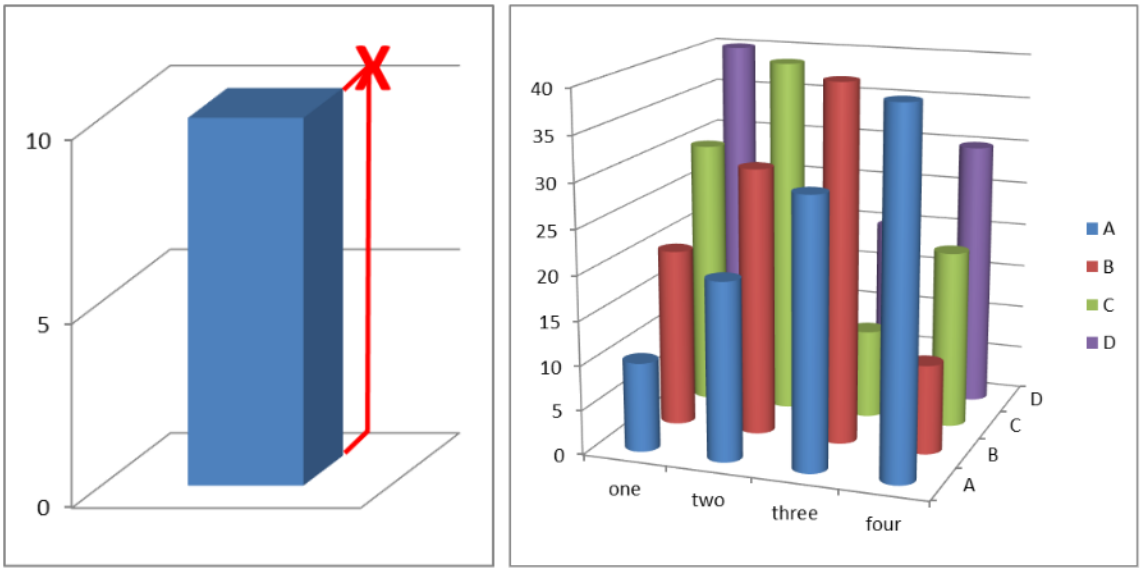


Figure 12: 3D 차트에서 데이터 값을 읽는 것은 어렵습니다.

4.2 표의 요소

표는 그래픽과 마찬가지로 효율적인 시각화 수단이 될 수 있습니다. 또한 제대로 제시되지 않으면 오해를 불러일으키거나 읽기 번거로울 수도 있습니다. 특히 데이터 포인트 수가 적거나 정확한 숫자를 보여줘야 하는 경우 표를 고려해야 합니다. 예를 들어 가독성과 해석을 돕기 위해 그래픽 위에 숫자를 겹쳐서 표시해야 하는 경우(예: 막대 차트의 막대 끝에 숫자 추가), 그래픽을 완전히 생략하고 숫자를 표로 표시하는 것을 고려할 수 있습니다.

표에는 그래픽과 마찬가지로 디자인 요소가 있습니다. 그중 일부는 다음에서 논의됩니다.

레이아웃

표의 한 가지 디자인 요소는 레이아웃, 즉 행과 열입니다. 레이아웃은 독자를 염두에 두고 적극적으로 선택해야 합니다. 일반적으로 수평보다 수직으로 숫자를 비교하는 것이 더 쉽습니다.

자릿수

자릿수는 표의 행이나 열(즉, 동일한 변수) 내에서는 일관되어야 하지만 변수 간에는 다를 수 있습니다(예: 표에 작은 숫자와 큰 숫자가 포함된 경우). 불필요한 정밀도는 피해야 하며 자릿수를 줄이면 읽기와 비교가 용이해질 수 있습니다.

정렬

큰 숫자와 작은 숫자를 쉽게 식별할 수 있도록 숫자는 오른쪽 정렬되어야 합니다. 이렇게 하면 큰 숫자가 말 그대로 튀어나오게 됩니다.

표 셀의 여러 숫자

표 셀에 단일 값 이상의 값(예: 평균 및 신뢰 구간)이 포함된 경우 데이터를 쉽게 읽을 수 있도록 표시하는 더 나은 방법이 있는지 고려해야 합니다. 옵션에는 숫자를 여러 표 셀, 행 또는 열로 나누거나 덜 중요한 숫자의 글꼴 크기를 줄이는 것(예: 평균이 신뢰 구간보다 더 중요할 수 있음)이 있습니다.

방향

피할 수 없는 경우가 아니라면 가로 방향은 권장하지 않습니다. 일부를 읽기 위해 문서를 회전하는 것은 인쇄물이든 화면이든 독자에게 번거로운 일입니다. 표를 둘로 나누는 것은 가로 방향을 피하는 해결책이 될 수 있지만, 표를 나누는 것이 중요한 비교를 더 어렵게 만드는지 고려해야 합니다.

글꼴 및 색상

특정 값을 강조하는 것과 같은 목적에 부합한다면 다른 글꼴과 색상을 사용할 수 있습니다. 장식 목적으로 다른 글꼴이나 색상을 사용하는 것은 권장하지 않습니다.

5 시각화 유형 선택하기

5.1 목표와 청중

어떤 유형의 데이터 시각화를 만들지 결정하는 과정은 간단한 질문에서 시작됩니다. 나는 왜 이것을 하고 있는가? 데이터 시각화는 목적을 충족해야 하므로 차트 유형, 디자인, 레이아웃 등에 대한 모든 결정은 그래픽의 의도된 목적에 대한 명확한 이해에서 비롯되어야 합니다.

볼로냐 대학교의 통계학 교수인 Christian Hennig은 다음 질문들을 검토할 것을 제안합니다:

1. 그래프의 목적이 무언가를 발견하는 것인가("분석 그래프"), 아니면 다른 사람들에게 주장하는 것인가?
2. 당신이 알아내고 싶은 것은 무엇인가?
3. 그래프의 청중은 누구인가? (자기 자신일 수도 있다.)

초기 단계에서 타겟 청중을 식별하는 것은 매우 중요합니다. 교육 수준, 기술적 전문성, 데이터 시각화 형식에 대한 이전 경험 및 기타 요인에 따라 다양한 그룹의 사람들이 서로 다른 수준의 그래프 리터러시를 가질 가능성이 높기 때문입니다. 의도한 청중의 요구를 제대로 고려하지 않은 디자인 결정은 목표를 달성하는 데 실패할 것입니다.

체크리스트

디자인 결정을 내리기 전에:

- 데이터 시각화의 의도된 목적을 명확히 하십시오.
- 타겟 청중의 요구와 다양한 시각화 유형에 대한 친숙도를 이해하십시오.

5.2 데이터

소프트웨어 도구는 일반적으로 사용자가 선택한 형식으로 데이터를 시각화할 수 있는 유연성을 제공합니다. 그러나 차트 유형에 대한 제안은 당면한 데이터 종류나 관심 있는 데이터 관계에 의해 결정되어야 합니다. 그런 다음, 앞서 논의한 대로 청중의 요구에 따라 형식 선택을 더 구체화해야 합니다.

시각화 유형을 결정하는 데 도움이 되는 다양한 온라인 도구가 있습니다. 다음을 추천합니다:

유형

From Data to Viz는 선택한 데이터 유형(수치형, 범주형 등)에 따라 다양한 추천 차트 형식으로 이어지는 일련의 의사 결정 트리를 사용자에게 제시합니다.

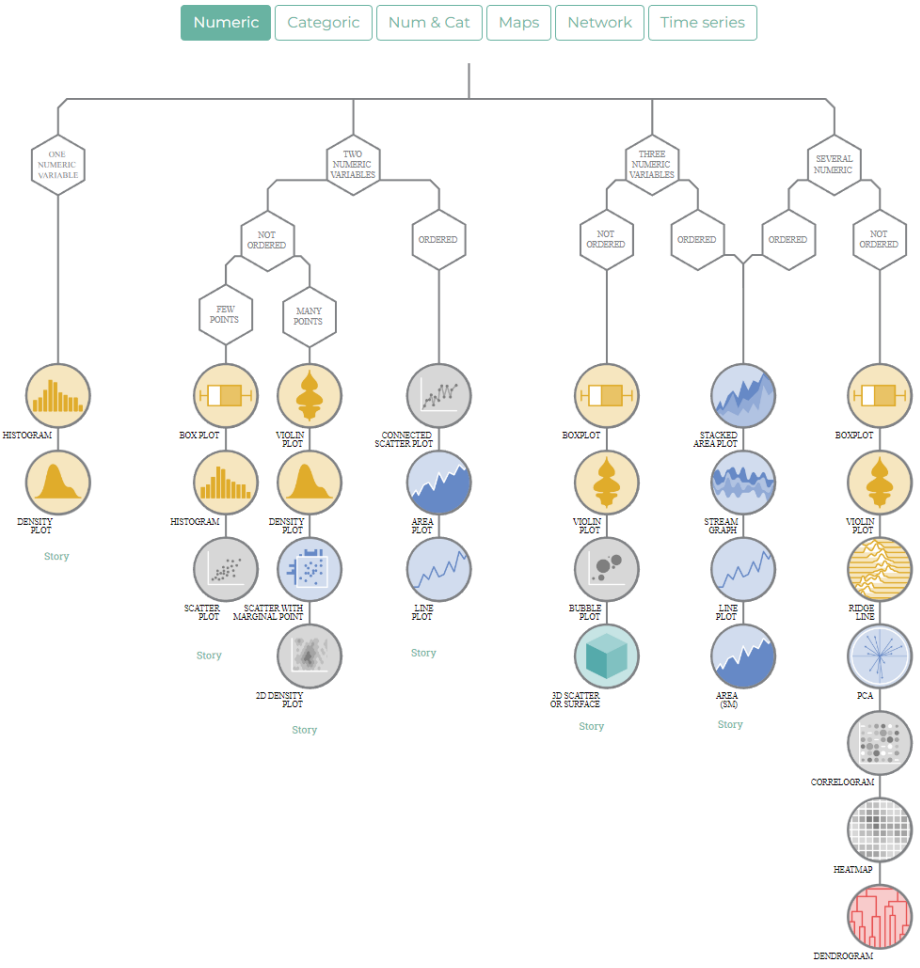


Figure 13: From Data to Viz의 스크린샷.

관계

Visual Vocabulary는 파이낸셜 타임즈(Financial Times)의 데이터 시각화 팀이 개발한 사이트입니다. 이를 통해 사용자는 "당신의 이야기에서 가장 중요한" 데이터 관계에 따라 차트 유형 선택의 폭을 좁힐 수 있습니다. 관계 옵션에는 편차, 상관관계, 시간 대비 변화, 순위 등이 포함됩니다.

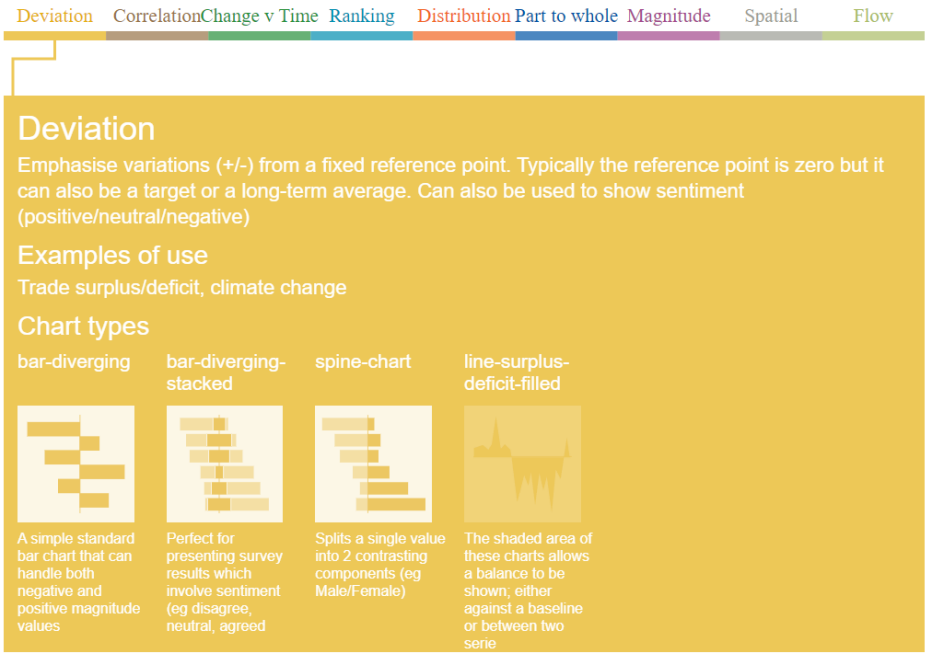


Figure 14: Visual Vocabulary의 스크린샷.

체크리스트

- 데이터 유형 또는 데이터 관계에 따라 차트 유형을 선택하십시오.
- 적절한 차트 유형 범위에서 선택할 때 타겟 청중의 요구와 예상되는 그래프 리터러시 수준을 염두에 두십시오.

> **Note**
>
> Few의 Show Me the Numbers (Few 2004) 및 Robbins의 Creating Better Graphs (Robbins 2006)에서 더 일반적인 지침을 제공합니다.

6 접근성을 위한 차트 스타일링

6.1 차트 스타일링 원칙

아래 두 차트는 동일한 데이터를 동일한 유형의 차트(수평 막대 차트로 시각화된 기니피그 치아 성장 데이터)를 사용하여 보여줍니다. 그러나 두 차트의 명확성은 확연히 다릅니다. 색상 선택, 텍스트 주석 추가, 글꼴 크기 변경, 더 유익한 레이블 등은 오른쪽 차트를 해석하기 훨씬 쉽게 만듭니다.[1] 이 섹션에서는 이러한 각 요소를 자세히 살펴보고, 데이터 시각화의 접근성과 해석 가능성을 높이기 위해 차트의 다양한 요소를 스타일링하는 방법에 대해 설명합니다.

```
library(ggplot2)
library(dplyr)
library(ggtext)
plot_data <- ToothGrowth %>%
```

[1] 동일한 치아 성장 데이터에 대한 다른 프레젠테이션은 GitHub 리포지토리의 이 토론(코드 포함)을 참조하십시오.

```r
  mutate(dose = factor(dose)) %>%
  group_by(dose, supp) %>%
  summarise(len = mean(len)) %>%
  ungroup()

# Unstyled plot
ggplot(
  data = plot_data,
  mapping = aes(x = len, y = dose, fill = supp)
) +
  geom_col(position = "dodge")

# Styled plot
ggplot(
  data = plot_data,
  mapping = aes(x = len, y = dose, fill = supp)
) +
  geom_col(
    position = position_dodge(width = 0.7),
    width = 0.7
  ) +
  scale_x_continuous(
    limits = c(0, 30),
    name = "    "
  ) +
  geom_text(
    mapping = aes(label = round(len, 0)),
    position = position_dodge(width = 0.7),
    hjust = 1.5,
    size = 6,
    fontface = "bold",
    colour = "white"
  ) +
  scale_fill_manual(values = c("#9B1D20", "#3D5A80")) +
  labs(
    title = "    ",
    subtitle = "60         C 3     (0.5, 1, 2 mg/ )
                        :
       <span style='color: #9B1D20'>**    **</span>
    <span style='color: #3D5A80'>**    **</span>.",
    y = "   (mg/ )"
  ) +
  theme_minimal(base_size = 14) +
  theme(
    legend.position = "none",
    plot.title = element_textbox_simple(face = "bold"),
    plot.subtitle = element_textbox_simple(
      margin = margin(t = 10),
      lineheight = 1.5
    ),
    plot.title.position = "plot",
```

```
    plot.margin = margin(15, 10, 10, 15),
    panel.grid = element_blank(),
    axis.text.x = element_blank()
  )
```

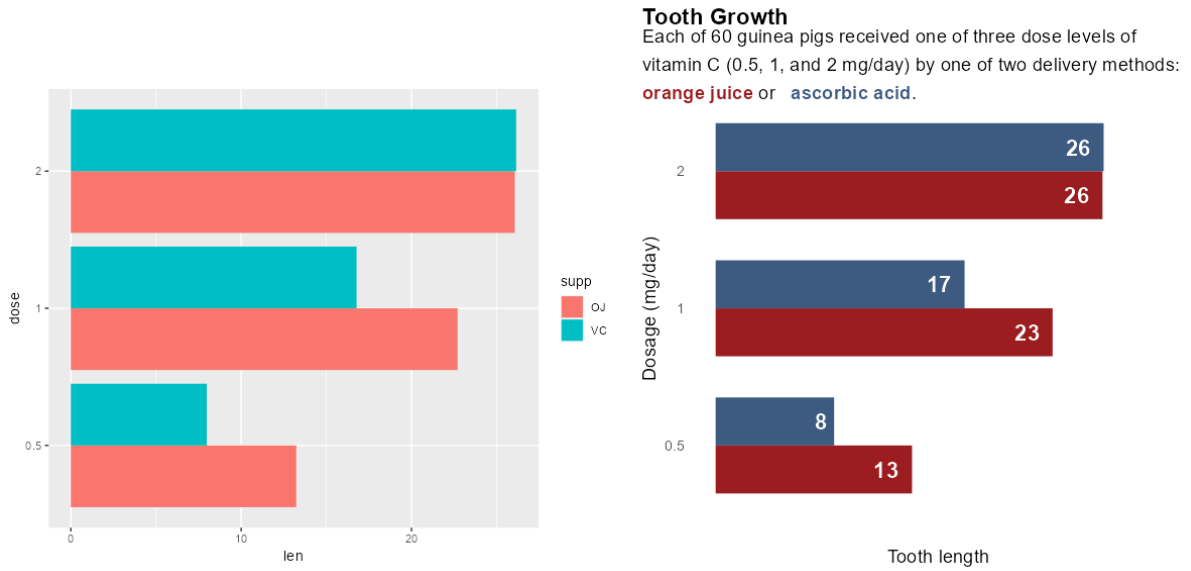


Figure 15: 기니피그 치아 성장을 시각화한 전후 막대 차트.

6.2 색상

색상 선택은 데이터 시각화의 접근성에 큰 영향을 미칠 수 있습니다. 색상을 올바르게 사용하면 전달하려는 이야기를 강조하는 데에도 도움이 될 수 있습니다.

색상을 선택하기 전에 스스로에게 질문해 보십시오. 여기서 정말 색상을 사용해야 하나? Beecham et al. (2021) 은 변수 간의 차이를 시각적으로 전달하는 데 있어 색상 사용이 가장 효과적이지 않은 방법 중 하나임을 보여주었습니다.

```
library(dplyr)
library(ggplot2)
plot_data <- mtcars %>%
  mutate(car = rownames(mtcars))

# Colour all bars
ggplot(
  data = plot_data,
  mapping = aes(
    y = reorder(car, disp),
    x = disp,
    fill = car
  )
) +
  geom_col() +
  labs(
```

```
    x = "  1",
    y = ""
  ) +
  coord_cartesian(expand = FALSE) +
  theme_minimal(base_size = 14) +
  theme(
    legend.position = "none",
    legend.title = element_blank(),
    plot.title = element_text(
      face = "bold",
      margin = margin(b = 10)
    ),
    plot.title.position = "plot",
    plot.margin = margin(15, 10, 10, 15)
  )

# Highlight one bar
ggplot(
  data = plot_data,
  mapping = aes(
    y = reorder(car, disp),
    x = disp,
    fill = (car == "Maserati Bora")
  )
) +
  geom_col() +
  scale_fill_manual(values = c("#AFE1AF", "#7a9d7a")) +
  labs(
    x = "  1",
    y = ""
  ) +
  coord_cartesian(expand = FALSE) +
  theme_minimal(base_size = 14) +
  theme(
    legend.position = "none",
    legend.title = element_blank(),
    plot.title = element_text(
      face = "bold",
      margin = margin(b = 10)
    ),
    plot.title.position = "plot",
    plot.margin = margin(15, 10, 10, 15)
  )
```

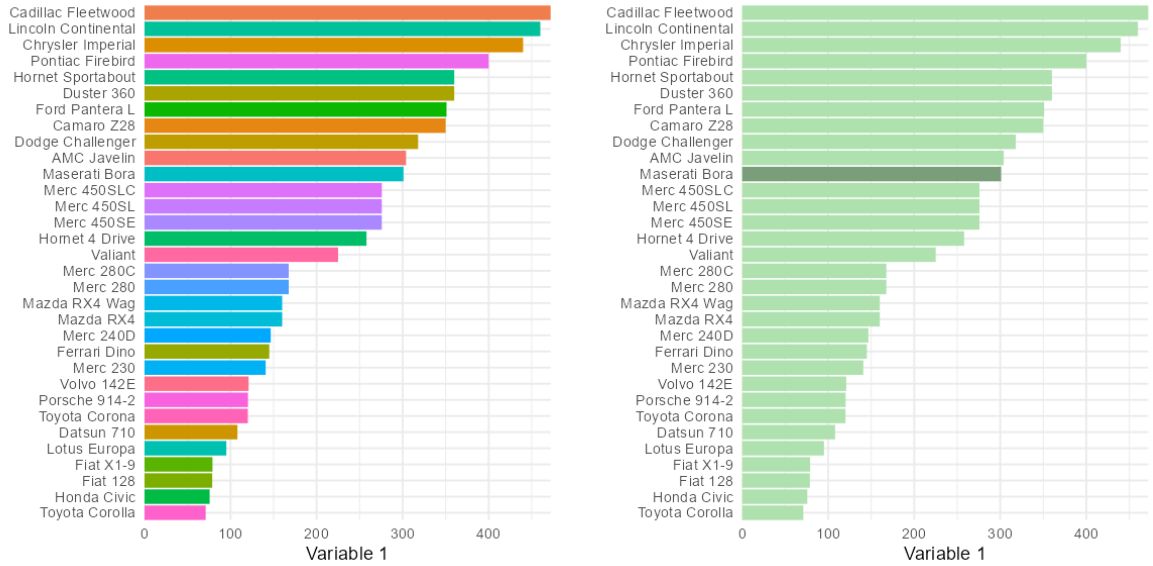


Figure 16: 색칠 대 강조 표시를 보여주는 자동차 데이터 시각화 막대 차트 전후.

위의 막대 차트 예시에서 y축의 정보를 복제하여 각 막대에 다른 색상을 칠하는 대신, 더 나은 접근 방식은 색상을 사용하여 데이터의 특정 요소(예: 특정 자동차)를 강조 표시하여 독자의 주의를 주장하려는 요점에 바로 집중시키는 것입니다.

서로 다른 그룹의 데이터 포인트를 구별하는 유일한 요소로 색상에 의존하지 않는 것이 좋습니다. 예를 들어 산점도에서 두 그룹의 포인트를 녹색 또는 주황색으로 칠하는 것 외에도 포인트를 원과 삼각형으로 인코딩할 수 있습니다.

```
library(readr)
library(dplyr)
library(tidyr)
library(ggplot2)
wheels <- read_csv(
  "https://raw.githubusercontent.com/rfordatascience/tidytuesday/master/data/2022/2022-08-09
  )
plot_data <- wheels %>%
  select(country, height, diameter) %>%
  drop_na() %>%
  filter(country %in% c("USA", "Japan"))

# Colour only
ggplot(
  data = plot_data,
  mapping = aes(
    x = diameter,
    y = height,
    colour = country
  )
) +
  geom_point(size = 3, alpha = 0.8) +
  scale_x_continuous(limits = c(0, 800)) +
  scale_y_continuous(limits = c(0, 800)) +
```

```
  scale_colour_brewer(palette = "Dark2") +
  coord_cartesian(expand = FALSE) +
  labs(
    title = "   ",
    x = "   (ft)",
    y = "   (ft)"
  ) +
  theme_minimal(base_size = 14) +
  theme(
    legend.position = "top",
    legend.title = element_blank(),
    plot.title = element_text(
      face = "bold",
      margin = margin(b = 10)
    ),
    plot.title.position = "plot",
    plot.margin = margin(15, 10, 10, 15)
  )

# Shapes and colours
ggplot(
  data = plot_data,
  mapping = aes(
    x = diameter,
    y = height,
    colour = country,
    shape = country
  )
) +
  geom_point(size = 3, alpha = 0.8) +
  scale_x_continuous(limits = c(0, 800)) +
  scale_y_continuous(limits = c(0, 800)) +
  scale_colour_brewer(palette = "Dark2") +
  coord_cartesian(expand = FALSE) +
  labs(
    title = "   ",
    x = "   (ft)",
    y = "   (ft)"
  ) +
  theme_minimal(base_size = 14) +
  theme(
    legend.position = "top",
    legend.title = element_blank(),
    plot.title = element_text(
      face = "bold",
      margin = margin(b = 10)
    ),
    plot.title.position = "plot",
    plot.margin = margin(15, 10, 10, 15)
  )
```

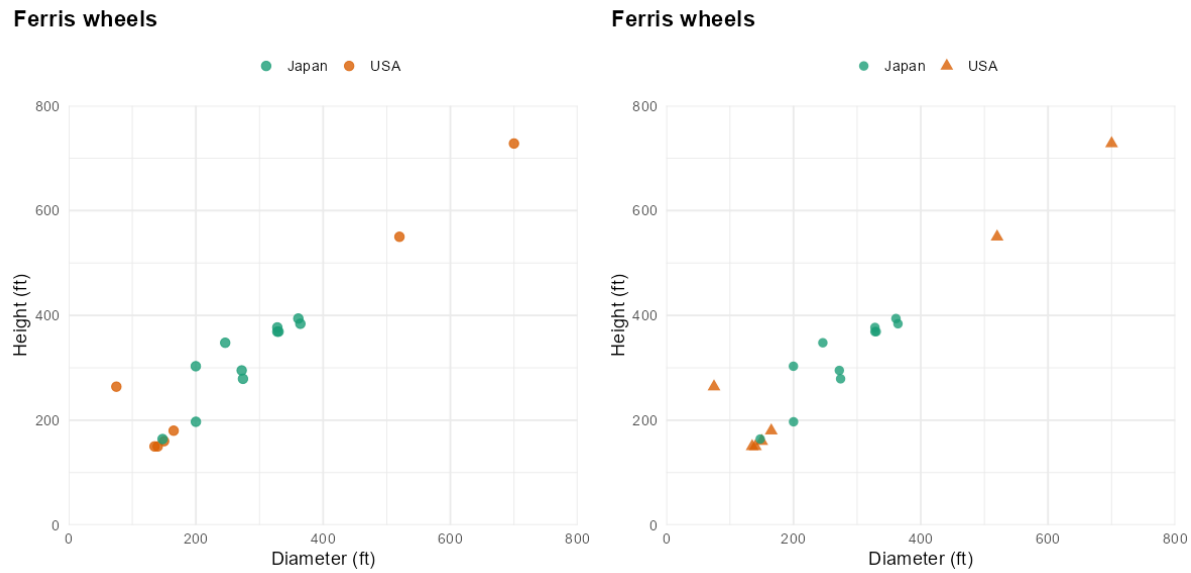


Figure 17: 점만 보여주는 것과 그룹을 구별하기 위해 점과 삼각형을 보여주는 대관람차 데이터 시각화 산점도 전후.

비효율적인 사용으로 인해 발생할 수 있는 어려움에도 불구하고 색상은 범주를 구별하거나 값의 변화를 보여주는 가장 일반적인 방법 중 하나입니다. 색상을 사용하기로 결정했다면 어떤 색상을 사용할지 결정해야 합니다.

색상 팔레트 유형

색상 팔레트는 일반적으로 세 가지 범주 중 하나에 속하며, 선택하는 색상 팔레트 유형은 시각화하려는 데이터 유형을 반영해야 합니다.

- **순차적(Sequential)**: 낮은 것에서 높은 것(또는 그 반대)으로 순서가 있는 데이터를 시각화하는 데 사용됩니다(예: 온도).
- **발산적(Diverging)**: 중요한 중간점에서 멀어지며 순서가 있는 데이터를 시각화하는 데 사용됩니다(예: 양방향으로 평균 온도보다 높거나 낮은 일수).
- **정성적(Qualitative)**: 범주 간의 차이 크기와 순서가 의미가 없는 범주형 데이터를 시각화하는 데 사용됩니다(예: 서로 다른 철도 노선).

```
library(ggplot2)
library(PrettyCols)

# Sequential
ggplot(
  data = data.frame(x = 1:7, y = 1),
  mapping = aes(x = x, y = y, fill = x)
) +
  geom_tile() +
  labs(title = "  (Sequential)") +
  scale_fill_pretty_c("Teals") +
  theme_void() +
  theme(
    legend.position = "none",
```

```
    plot.title = element_text(size = 20, face = "bold"),
    plot.margin = margin(15, 10, 10, 15)
  )

# Diverging
ggplot(
  data = data.frame(x = 1:7, y = 1),
  mapping = aes(x = x, y = y, fill = x - mean(x))
) +
  geom_tile() +
  labs(title = "  (Diverging)") +
  scale_fill_gradient2(low = "#f1a340", high = "#998ec3") +
  theme_void() +
  theme(
    legend.position = "none",
    plot.title = element_text(size = 20, face = "bold"),
    plot.margin = margin(15, 10, 10, 15)
  )

# Qualitative
ggplot(
  data = data.frame(x = 1:7, y = 1),
  mapping = aes(x = x, y = y, fill = factor(x))
) +
  geom_tile() +
  labs(title = "  (Qualitative)") +
  scale_fill_brewer(palette = "Dark2") +
  theme_void() +
  theme(
    legend.position = "none",
    plot.title = element_text(size = 20, face = "bold"),
    plot.margin = margin(15, 10, 10, 15)
  )
```

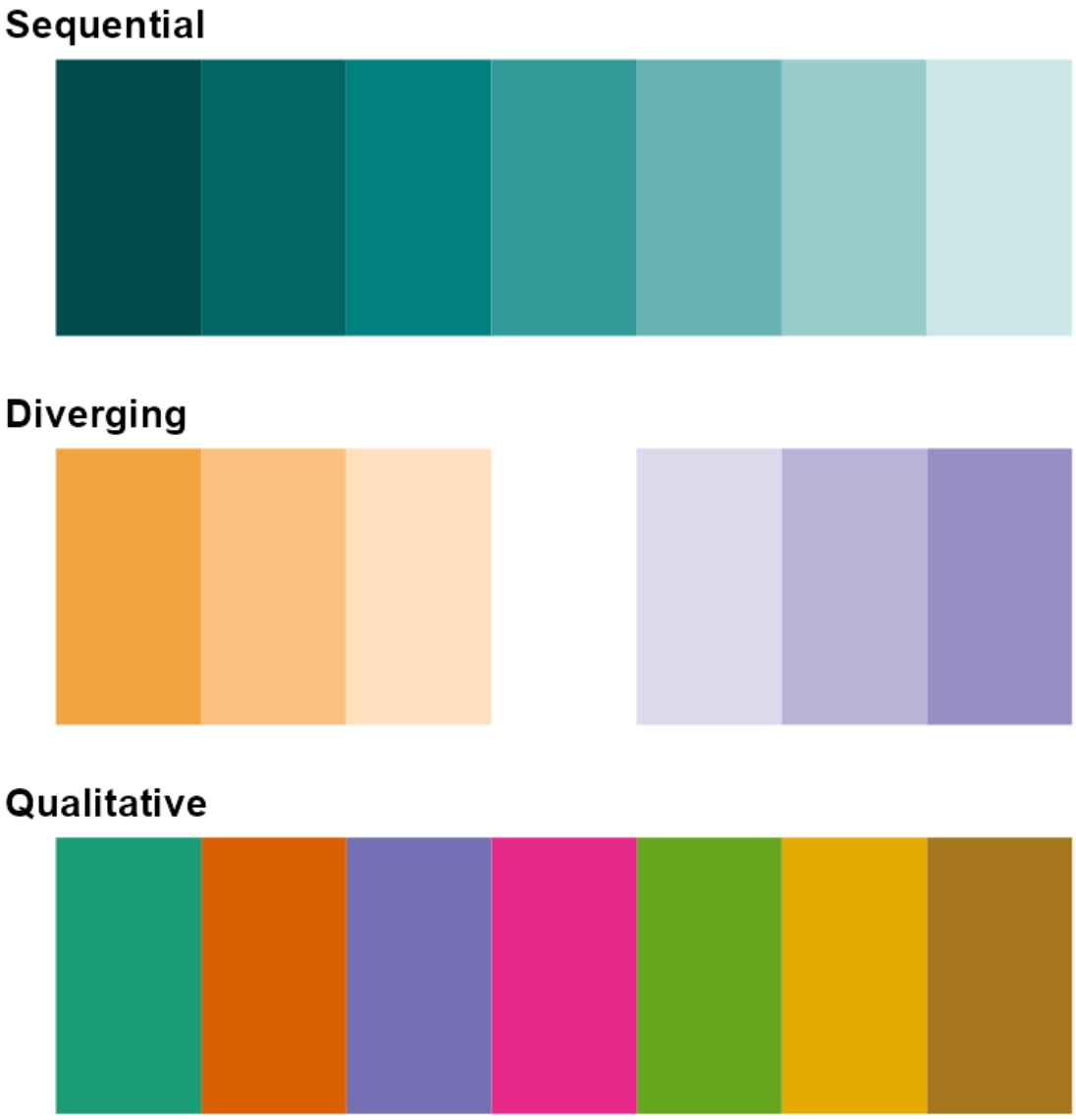


Figure 18: 다양한 유형의 색상 팔레트.

접근 가능한 색상 팔레트

특정 색상 팔레트를 선택할 때 고려해야 할 몇 가지 중요한 요소가 있습니다.

색상은 모든 독자에게 서로 구별되어야 합니다. 일부 색상을 구별할 수 없게 만드는 여러 가지 형태의 색맹이 있습니다. 색맹 검사기 (“Coblis — Color Blindness Simulator” n.d.)를 사용하면 다양한 유형의 색맹 하에서 플롯이 어떻게 보일지 확인할 수 있습니다. 이는 색조의 차이를 사용하여 서로 다른 값을 구별하는 발산적 및 정성적 팔레트에서 특히 중요합니다. 반면 순차적 팔레트는 값이 어떻게 변하는지 보여주기 위해 광도(luminosity)를 사용하는 경우가 많습니다. Tennekes and Puts (2023) 은 단일 색조 팔레트가 일반적으로 색맹 친화적일 가능성이 더 높기 때문에 더 안전한 선택이라고 지적합니다. 또한 Tennekes and Puts (2023) 은 광도와 채도(색상의 순도)의 변동성이 클수록 색맹이 있는 사람들이 색상을 구별할 가능성이 더 높다고 지적합니다. Paul Tol (2021)은 색상 선택에 대한 유용한 리소스와 몇 가지 추천 팔레트를 제공합니다.

```r
library(ggplot2)
library(dplyr)
library(ggtext)
library(colorblindr)
plot_data <- ToothGrowth %>%
  mutate(dose = factor(dose)) %>%
  group_by(dose, supp) %>%
  summarise(len = mean(len)) %>%
  ungroup()

g <- ggplot(
  data = plot_data,
  mapping = aes(x = len, y = dose, fill = supp)
) +
  geom_col(
    position = position_dodge(width = 0.7),
    width = 0.7
  ) +
  scale_x_continuous(
    limits = c(0, 30),
    name = "    "
  ) +
  geom_text(
    mapping = aes(label = round(len, 0)),
    position = position_dodge(width = 0.7),
    hjust = 1.5,
    size = 6,
    fontface = "bold",
    colour = "white"
  ) +
  scale_fill_manual(values = c("#9B1D20", "#3D5A80")) +
  labs(
    title = "    ",
    subtitle = "60         C 3     (0.5, 1, 2 mg/ )
                        :
       <span style='color: #9B1D20'>**    **</span>
    <span style='color: #3D5A80'>**    **</span>.",
    y = "   (mg/ )"
  ) +
  theme_minimal(base_size = 14) +
  theme(
    legend.position = "none",
    plot.title = element_textbox_simple(face = "bold"),
    plot.subtitle = element_textbox_simple(
      margin = margin(t = 10),
      lineheight = 1.5
    ),
    plot.title.position = "plot",
    plot.margin = margin(15, 10, 10, 15),
    panel.grid = element_blank(),
    axis.text.x = element_blank()
```

```
  )

cvd_grid(g)
```

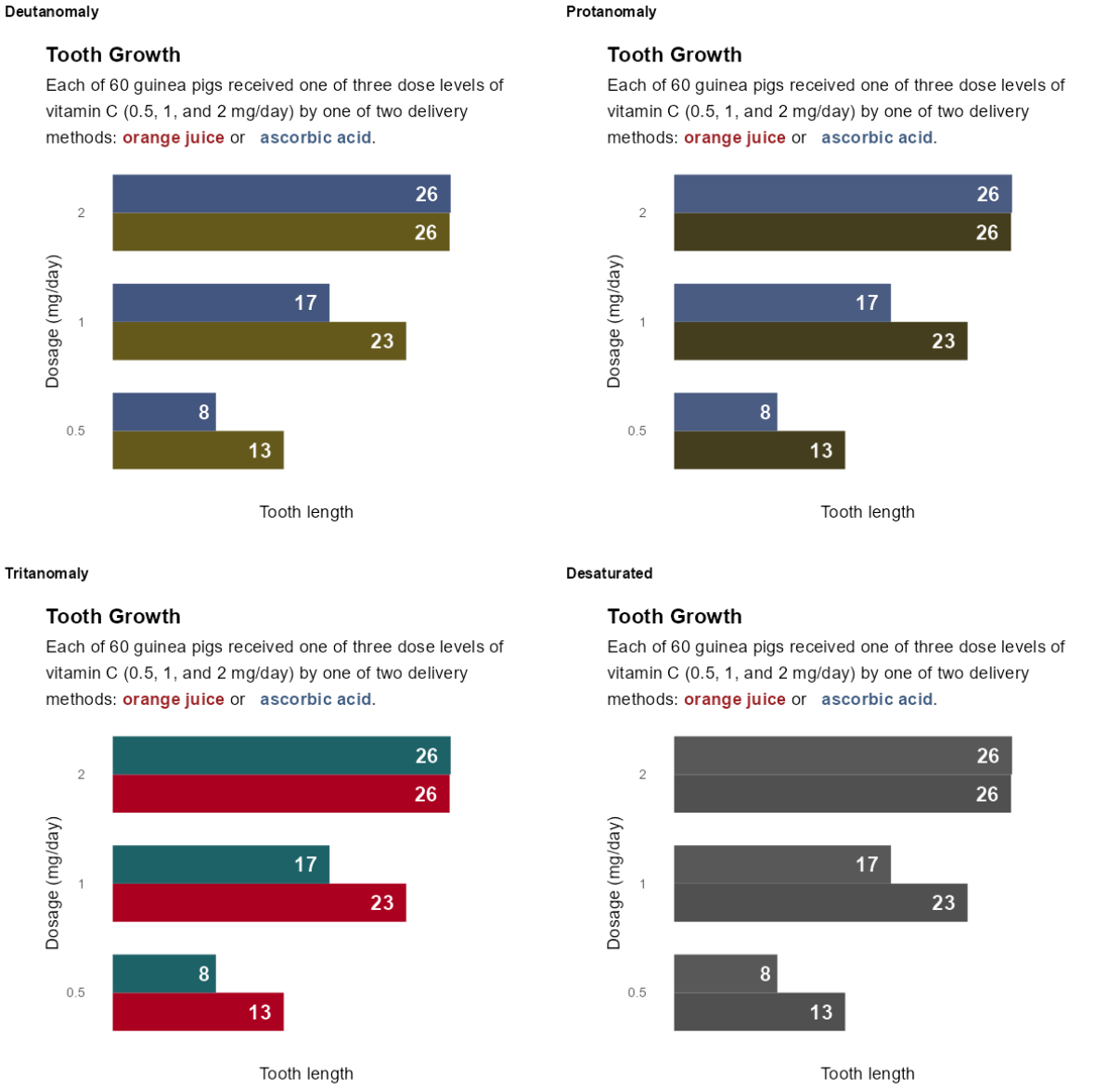


Figure 19: 다양한 유형의 색맹에서 동일한 막대 차트의 시뮬레이션된 모양.

흑백으로 인쇄했을 때 색상은 서로 구별되어야 합니다. 독자 또는 학술지는 시각화를 단색으로 인쇄하도록 선택할 수 있습니다. 시각적으로 뚜렷해 보이는 색상이라도 광도(색조가 얼마나 밝거나 어두운지를 나타내는 척도)가 비슷하면 흑백으로 인쇄했을 때 구별할 수 없게 보일 수 있습니다.

색상은 데이터와 적절하게 일치해야 합니다. 예를 들어 여성과 남성 관련 데이터에 분홍색과 파란색을 선택하는 것과 같은 고정관념에 빠지지 않는 것이 중요합니다. 일반적인 색상 연관성을 뒤집어서(예: 좋음은 빨간색, 나쁨은 녹색) 독자를 혼란스럽게 하지 마십시오. Muth (2018) 은 성별 데이터를 표시하기 위한 색상 사용에 대해 논의하며, 많은 요점은 다른 유형의 데이터로 확장될 수 있습니다.

체크리스트

- 사용되는 다른 색상의 수를 최소화하십시오.
- 색상이 색맹 친화적인지, 흑백으로 볼 때 충분한 대비가 있는지 확인하십시오.
- 색상 팔레트 유형이 시각화된 데이터 유형과 일치하는지 확인하십시오.
- 가능한 경우 색상이 아닌 요소를 사용하여 다른 그룹을 구별하십시오.
- 고정관념과 색상에 대한 혼란스러운 연관성을 피하십시오.

6.3 주석

데이터 시각화에 주석을 추가한다는 것은 독자가 그래픽의 요점을 이해하는 데 도움이 되도록 설명이나 해명을 추가하는 텍스트를 넣는 것을 의미합니다. 주석은 세부 정보나 설명을 추가하거나, 흥미로운 데이터 포인트를 강조하거나, 차트를 어떻게 해석해야 하는지 명확히 하는 데 사용할 수 있습니다. 추가 텍스트는 독자에게 매우 유용할 수 있지만, 텍스트가 너무 많으면 주의를 산만하게 하고 요점에 집중하기 어렵게 만들 수 있으므로 데이터 시각화에 텍스트를 과부하하지 않는 것이 중요합니다.

데이터 시각화의 명확성을 높이는 주석의 일반적인 사용법은 꺾은선 그래프나 막대 차트의 데이터 포인트에 직접 레이블을 지정하는 것입니다. 꺾은선 그래프에서는 선 끝의 값이 y축 레이블(기본적으로 왼쪽)에서 멀리 떨어져 있는 경우가 많으므로 선 끝에 값을 표시하는 것이 일반적입니다. 막대 차트에서는 각 막대 끝이나 근처에 각 막대의 값을 추가하면 독자가 직접 값을 찾아볼 필요가 없습니다. 이렇게 하면 독자가 정확한 값을 찾기 위해 눈을 움직여야 하는 양을 줄이고 더 좋고 정확한 비교를 할 수 있습니다. 예를 들어 이러한 값이 표에 추가로 제공되는 경우에는 항상 필요한 것은 아닙니다.

```
library(ggplot2)
library(dplyr)
plot_data <- ToothGrowth %>%
  mutate(dose = factor(dose)) %>%
  group_by(dose, supp) %>%
  summarise(len = mean(len)) %>%
  ungroup()

# Not annotated
ggplot(
  data = plot_data,
  mapping = aes(
    x = len,
    y = dose,
    fill = supp
  )
) +
  geom_col(
    position = position_dodge(width = 0.7),
    width = 0.7
  ) +
  scale_x_continuous(
    limits = c(0, 30),
    name = "    "
  ) +
  scale_fill_manual(
```

```
    name = "  : ",
    values = c("#9B1D20", "#3D5A80")
  ) +
  labs(
    title = "    ",
    y = "  "
  ) +
  theme_minimal(base_size = 14) +
  theme(
    legend.position = "top",
    plot.title = element_text(face = "bold"),
    plot.title.position = "plot",
    plot.margin = margin(15, 10, 10, 15)
  )

# Annotated
ggplot(
  data = plot_data,
  mapping = aes(
    x = len,
    y = dose,
    fill = supp
  )
) +
  geom_col(
    position = position_dodge(width = 0.7),
    width = 0.7
  ) +
  scale_x_continuous(
    limits = c(0, 30),
    name = "    "
  ) +
  geom_text(
    mapping = aes(label = round(len, 0)),
    position = position_dodge(width = 0.7),
    hjust = 1.5,
    size = 6,
    fontface = "bold",
    colour = "white"
  ) +
  scale_fill_manual(
    name = "  : ",
    values = c("#9B1D20", "#3D5A80")
  ) +
  labs(
    title = "    ",
    y = "  "
  ) +
  theme_minimal(base_size = 14) +
  theme(
    legend.position = "top",
```

```
    plot.title = element_text(face = "bold"),
    plot.title.position = "plot",
    plot.margin = margin(15, 10, 10, 15)
  )
```

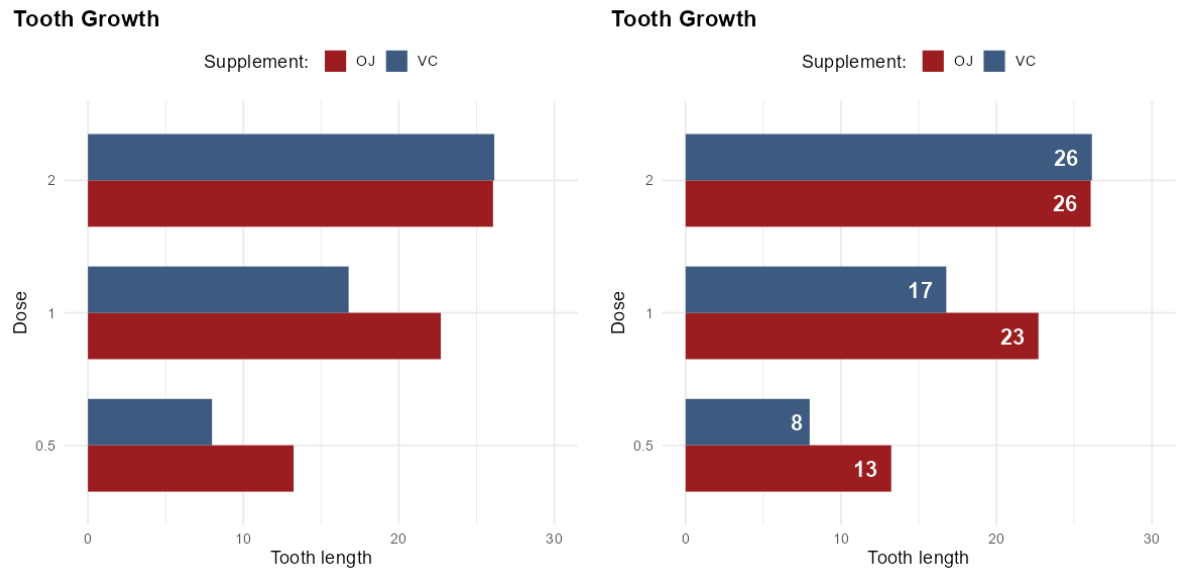


Figure 20: 막대 차트에 레이블을 지정하기 위해 주석을 추가하기 전과 후.

주석의 배치도 중요합니다:

- 산점도에서는 모든 데이터 포인트에 직접 레이블을 지정하는 것은 일반적으로 선택 사항이 아닙니다. 왜냐하면 포인트가 너무 많아서 혼란스러울 수 있기 때문입니다. 대신 이상치와 같이 흥미로운 포인트 한두 개만 강조 표시하도록 선택할 수 있습니다.
- 이상치나 관심 지점을 설명하기 위해 더 광범위한 텍스트를 추가할 때는 관련 위치에 텍스트를 배치하십시오. 예를 들어 2008년의 가치 감소를 설명하기 위해 꺾은선 차트에 주석을 달 경우 2008년 근처에 텍스트를 배치하십시오.
- 모든 공백을 텍스트로 채우지 마십시오. “Chart Titles and Text” (n.d.) 은 독자를 압도하지 않도록 차트당 최대 3~4개의 주석을 권장합니다. 이는 또한 주석을 간결하고 요점에 맞게 유지해야 함을 의미합니다. 추가적인 텍스트 설명이 이미 포함되어 있을 가능성이 높은 저널 간행물이나 잡지 기사에 수반되는 데이터 시각화의 경우 특히 그렇습니다.

체크리스트

- 추가 설명이 필요한 이상치나 직접 레이블을 지정할 수 있는 값이 있는지 확인하십시오.
- 레이블은 합리적으로 배치되어 있으며 데이터 시각화가 텍스트로 붐비지 않도록 하십시오.
- 텍스트와 배경 사이에 충분한 대비가 있는지 확인하십시오.

6.4 글꼴

글꼴 선택은 데이터 시각화를 이해하기 쉽게 만드는 핵심 구성 요소입니다. 어떤 글꼴은 다른 글꼴보다 읽기 쉬우며, 시각 장애가 있거나 난독증과 같은 학습 장애가 있는 사람들에게 특히 그렇습니다. 글꼴을 잘못 선택하면 상당수의 청중이 시각화에 접근하지 못할 수 있습니다.

사용하는 글꼴 유형의 수를 최소화하십시오. 고정된 크기 집합을 사용하고, 최대 두 가지 글꼴 패밀리를 사용하며, 글꼴 서체 혼합 사용을 최소화하십시오. 이 모든 것이 독자가 시각화를 볼 때 불필요한 작업을 줄이는 데 도움이 됩니다. 색상 선택과 관련된 이전 조언은 여기에도 관련이 있습니다. 배경색과 텍스트 색상 사이에 충분한 대비가 있는지 확인하는 것이 특히 중요합니다.

글꼴 크기

큰 글꼴이 읽기 쉽습니다. 인쇄물이나 웹사이트의 경우 글꼴 크기가 최소 12pt 이상일 것을 일반적으로 권장합니다. 프레젠테이션을 만드는 경우 글꼴은 최소 36pt 이상이어야 방 뒤쪽에 있는 사람들도 볼 수 있습니다.

글꼴 패밀리

그렇다면 좋은 글꼴 패밀리 선택은 무엇일까요? 글꼴에는 세 가지 주요 유형이 있습니다:

- **세리프(Serif)**: 세리프는 일부 글꼴에서 발생하는 긴 획 끝의 작은 획입니다. 이러한 세리프가 있는 글꼴을 세리프 글꼴이라고 합니다. (바탕체 계열)
- **산세리프(Sans serif)**: 세리프가 없는 글꼴을 산세리프 글꼴이라고 합니다. (고딕체 계열)
- **모노스페이스(Monospace)**: 모노스페이스 글꼴은 각 문자가 동일한 너비를 차지하는 글꼴입니다.

어떤 유형의 글꼴(세리프, 산세리프 또는 모노스페이스)이 더 접근하기 쉬운지에 대한 합의는 없습니다. 산세리프 글꼴의 단순한 문자는 시각 장애가 있는 독자의 가독성을 높일 수 있는 반면, 난독증이 있는 독자는 문자를 구별하기 더 어렵다고 느낄 수 있습니다. 세리프 글꼴(예: Times New Roman)은 장식선이 문자의 모양을 산만하게 하므로 읽기가 더 어려울 수 있습니다. 화면의 픽셀화가 문자를 더욱 왜곡할 수 있는 디지털 간행물에서는 특히 그렇습니다. 온라인에 표시되는 이미지의 텍스트에는 세리프 글꼴을 사용하지 않는 것이 좋습니다. Arial, Calibri, Verdana와 같은 일반적인 산세리프 글꼴은 종종 접근하기 쉬운 것으로 간주됩니다.

Dyslexie 및 OpenDyslexic은 문자의 아래쪽 글꼴 두께를 늘려 문자가 움직이는 것처럼 보이는 현상을 줄임으로써 난독증이 있는 사람들의 가독성을 돕기 위해 특별히 설계된 글꼴입니다. Atkinson Hyperlegible은 Braille Institute of America에서 개발한 글꼴로, 저시력 독자를 위해 문자 간의 구별을 극대화하도록 설계되었습니다. 무료로 사용할 수 있으며 Braille Institute에서 다운로드하거나 Google Fonts를 통해 사용할 수 있습니다. 특별히 설계된 글꼴이 실제로 더 접근하기 쉬운지에 대해서는 상충되는 증거가 있습니다Wery and Diliberto (2017). 따라서 산세리프 글꼴을 고수하는 것이 더 안전한 선택일 가능성이 높습니다.

일부 글꼴, 특히 산세리프 글꼴은 다른 문자와 매우 유사하게 보이는 문자가 있습니다. 예를 들어 대문자 I, 소문자 l, 숫자 1은 때때로 구별할 수 없습니다. 각 문자와 숫자에 대해 구별되는 특징이 있는 글꼴 서체를 선택하십시오. Source Sans Pro와 Verdana는 모두 이 조건을 충족합니다.

글꼴 서체

굵게 또는 이탤릭 효과와 같은 글꼴 서체는 텍스트의 특정 부분을 강조하는 데 사용할 수 있습니다. 예를 들어 플롯 주석은 설명 텍스트 사이에 배치된 특정 값을 강조 표시할 수 있습니다. 이탤릭체 텍스트는 강조 목적으로 사용될 때 대문자 텍스트와 마찬가지로 일반적으로 읽기 더 어려운 것으로 간주됩니다(Rello and Baeza-Yates 2016). 대신 강조하기 위해 굵은 텍스트를 사용하십시오(드물게!).

체크리스트

보시다시피 모든 사람이 시각화에 접근할 수 있도록 보장하는 단일 글꼴 권장 사항은 없습니다. 그러나 접근성을 개선하기 위해 할 수 있는 몇 가지 작업이 있습니다.

- 글꼴 크기가 최소 12pt 이상인지 확인하십시오.
- 대문자 I, 소문자 l, 숫자 1이 구별되는지 확인하십시오.
- 사용되는 다른 글꼴의 수를 최소화하십시오(2개 이하).
- 이탤릭체와 대문자 텍스트의 사용을 최소화하고 대신 강조를 위해(드물게) 굵은 텍스트를 사용하십시오.

6.5 대체 텍스트

대체 텍스트(Alternative text의 줄임말)는 차트를 포함한 이미지의 시각적 측면과 목적을 설명하는 텍스트입니다. 대체 텍스트는 다양한 용도로 사용되지만 주된 목적은 스크린 리더가 대체 텍스트를 소리 내어 읽을 때 시각 장애가 있는 사용자가 이미지를 해석하도록 돕는 것입니다.

Green (2023) 에서 Mine Dogucu는 시각 장애가 있는 사람들이 차트의 콘텐츠를 놓치지 않도록 데이터 시각화에 대체 텍스트를 추가하는 것의 중요성에 대해 논의합니다. Cesal (2020) 은 데이터 시각화를 위한 대체 텍스트 작성을 돕는 간단한 구조를 제공합니다.

7 RSS 간행물을 위한 차트 스타일링

7.1 다양한 도구를 사용한 차트 스타일링

이 섹션에서는 데이터 시각화를 만드는 데 사용되는 가장 일반적인 소프트웨어 유형으로 구축된 차트의 스타일을 변경하는 방법에 대한 기술적 세부 정보를 설명합니다.

R

R(R Core Team 2021)은 통계 컴퓨팅 및 그래픽에 널리 사용되는 프로그래밍 언어입니다. R에는 데이터 시각화를 만드는 데 사용할 수 있는 패키지가 많으며 여기서는 모두 다루지 않습니다. 대신 가장 일반적인 방법인 내장 base R 그래픽과 {ggplot2} 패키지(Wickham 2016b)를 다룹니다.

저자가 이 문서의 지침에 맞게 차트 스타일을 지정하는 데 도움을 주기 위해 R 패키지 {RSSthemes}를 개발했습니다. 여기에 패키지 사용 예제를 포함하지만 독자들은 업데이트 사항에 대해 GitHub 리포지토리를 확인하는 것이 좋습니다.

다음 명령어를 사용하여 CRAN에서 {RSSthemes} 패키지를 설치할 수 있습니다:

```
install.packages("RSSthemes")
```

GitHub에서 개발 버전을 설치할 수 있습니다(CRAN 버전만큼 안정적일 것이라고 약속할 수는 없지만):

```
remotes::install_github("nrennie/RSSthemes")
```

그런 다음 다음을 사용하여 패키지를 로드할 수 있습니다:

```
library(RSSthemes)
```

Base R

R에는 사용자가 추가 패키지를 설치하지 않고도 광범위한 데이터 시각화를 만들 수 있는 내장 그래픽 기능이 있습니다. Jumping Rivers의 이 블로그 게시물 (“Styling Base r Graphics” 2018)은 base R에서 만든 그래픽 스타일을 지정하는 방법에 대한 지침을 제공합니다.

예: base R에서 막대 차트 색상 변경.

모든 막대, 선, 점 등이 같은 색상을 가져야 하는 경우 `col` 인수를 RSS 색상 중 하나로 설정할 수 있습니다. 옵션은 `signif_red`, `signif_blue`, `signif_green`, `signif_orange`, 또는 `signif_yellow`입니다.

```
barplot(table(mtcars$gear), col = signif_blue)
```

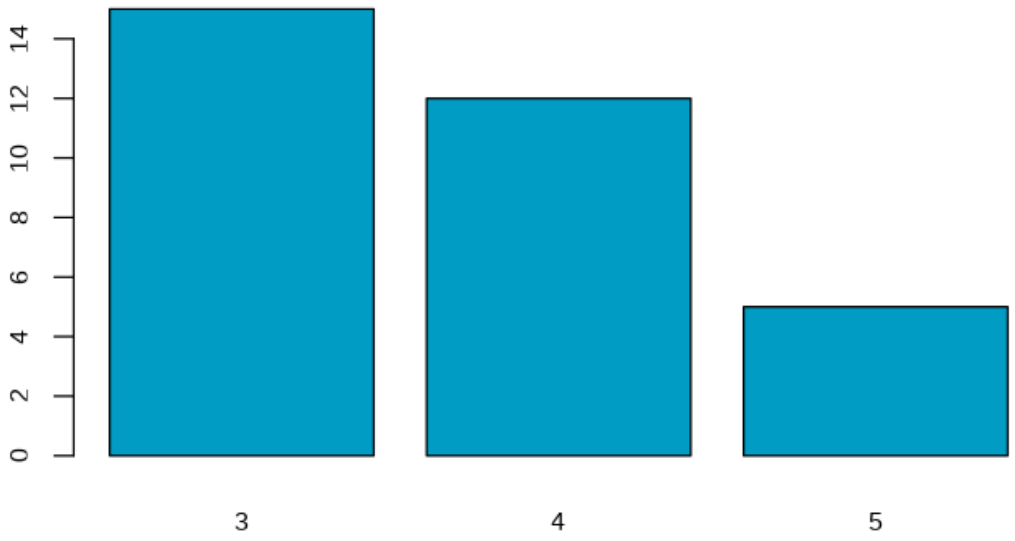


Figure 21: Significance 파란색으로 칠해진 막대가 있는 막대 차트.

플롯의 색상이 데이터의 값을 기반으로 하는 경우 `palette()` 함수를 사용하여 기본 색상을 설정할 수 있습니다. {RSSthemes} 내에서 `set_rss_palette()` 함수는 사용되는 기본 색상을 변경합니다. 현재 {RSSthemes}에는 세 가지 팔레트를 사용할 수 있지만 앞으로 더 추가할 예정입니다. 옵션은 `signif_qual`, `signif_div`, 또는 `signif_seq`입니다.

```
set_rss_palette("signif_qual")
plot(1:4, 1:4, col=1:4, pch=19, cex=3, xlab="", ylab="")
```

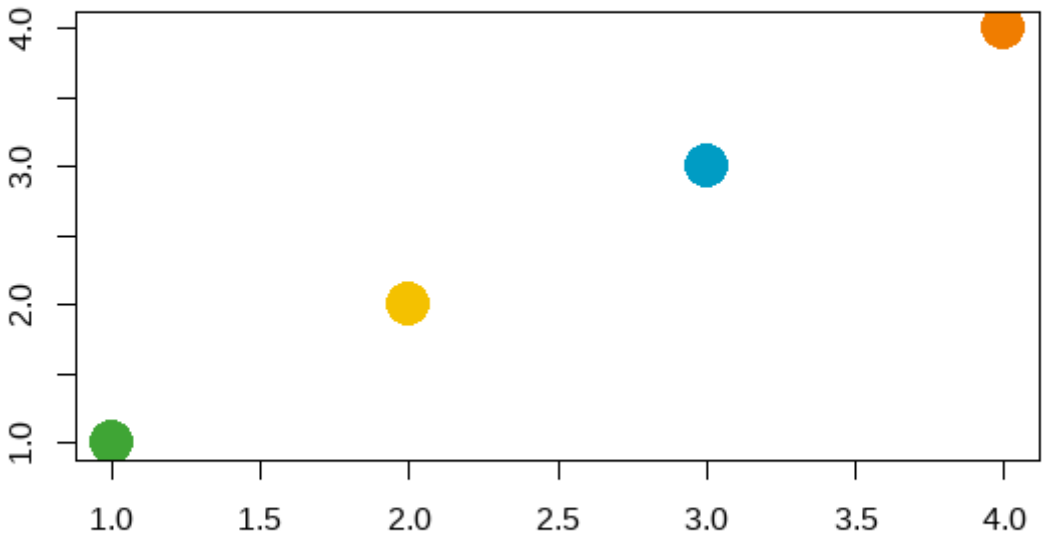


Figure 22: `signif_qual` 팔레트의 색상을 보여주는 산점도.

`palette("default")`를 실행하여 원래 base R 색상으로 재설정합니다.

예: base R 플롯의 스타일링 변경.

`plot()` 함수(및 `barplot()`, `hist()`와 같은 관련 base R 플로팅 함수) 내에는 플롯의 비 데이터 요소가 어떻게 보일지 제어하는 인수가 있습니다. 예를 들어 `family` 인수는 사용되는 글꼴 패밀리를 변경합니다. `par()` 함수를 호출하여 이러한 인수 중 상당수를 전역적으로 설정할 수도 있습니다. {RSSthemes} 내에는 모든 base R 플롯에 대한 텍스트 정렬 및 글꼴을 포함하여 몇 가지 기본 옵션을 설정하는 `set_signif_par()` 함수가 있습니다. `abline()` 함수를 사용하여 참조선을 추가하는 것도 좋습니다.

```
set_signif_par()
plot(1:4, 1:4, col=1:4, pch=19, cex=3, xlab="", ylab="",
     main = "  Significance  ",
     sub = " :     ")
abline(h=1:4, v=1:4, col = "lightgrey")
```

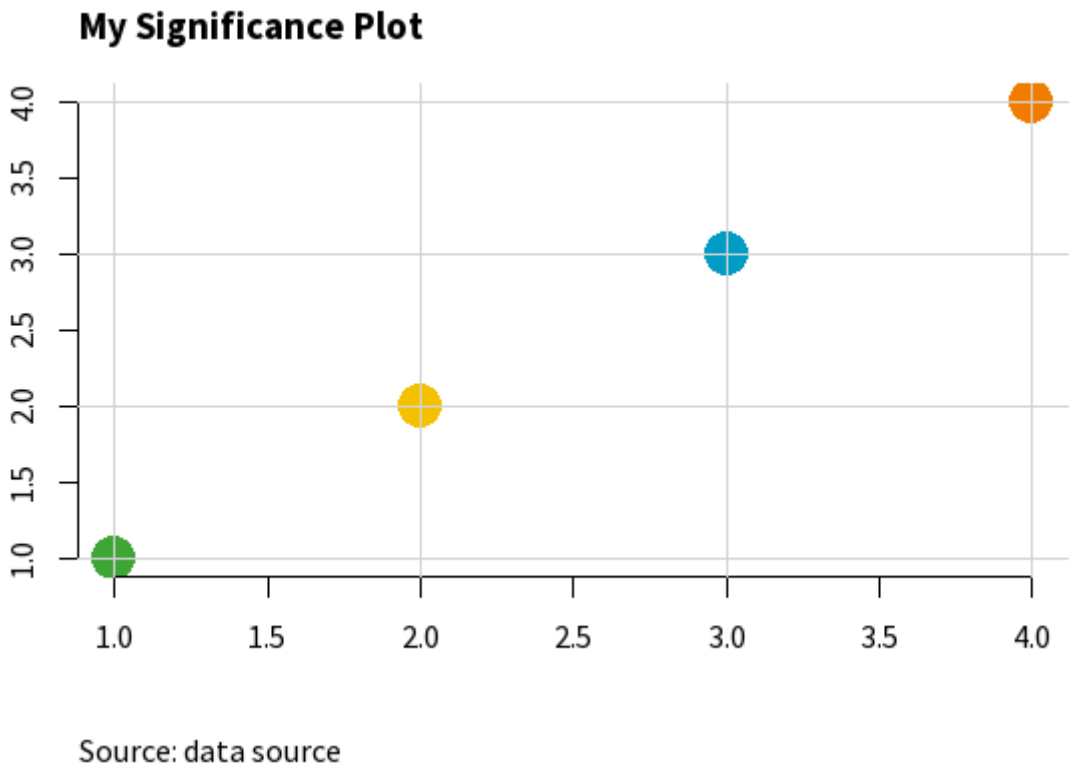


Figure 23: `set_signif_par()`에 의해 구현된 base R 스타일링을 보여주는 산점도.

{ggplot2}

{ggplot2}는 데이터 시각화를 만들기 위해 특별히 고안된 {tidyverse} 프레임워크 내의 R 패키지입니다. 패키지 문서는 다양한 유형의 차트를 만드는 방법에 대한 지침을 제공합니다. {ggplot2} 시각화의 색상 변경 및 스타일 변경에 대한 조언은 Hadley Wickham의 책 ggplot2: Elegant Graphics for Data Analysis (Wickham 2016b)에서 찾을 수 있습니다.

{ggplot2}로 플롯을 만들기 위해 기본 데이터 세트를 설정해 보겠습니다.

```
library(ggplot2)
plot_df <- data.frame(x = LETTERS[1:4],
                      y = 1:4)
```

예: {ggplot2}에서 매핑되지 않은 색상 변경.

{ggplot2}에서 `colour`(또는 `color`) 인수는 요소의 윤곽선을 나타내는 색상을 변경하고, `fill`은 요소를 채우는 색상을 변경합니다. 모든 막대, 선 또는 점이 동일한 색상을 가져야 하는 경우 `fill` 또는 `colour` 인수를 RSS 색상 중 하나로 설정할 수 있습니다. 옵션은 `signif_red`, `signif_blue`, `signif_green`, `signif_orange`, 또는 `signif_yellow`입니다.

```
ggplot(data = plot_df,
       mapping = aes(x = x, y = y)) +
  geom_col(fill = signif_yellow)
```

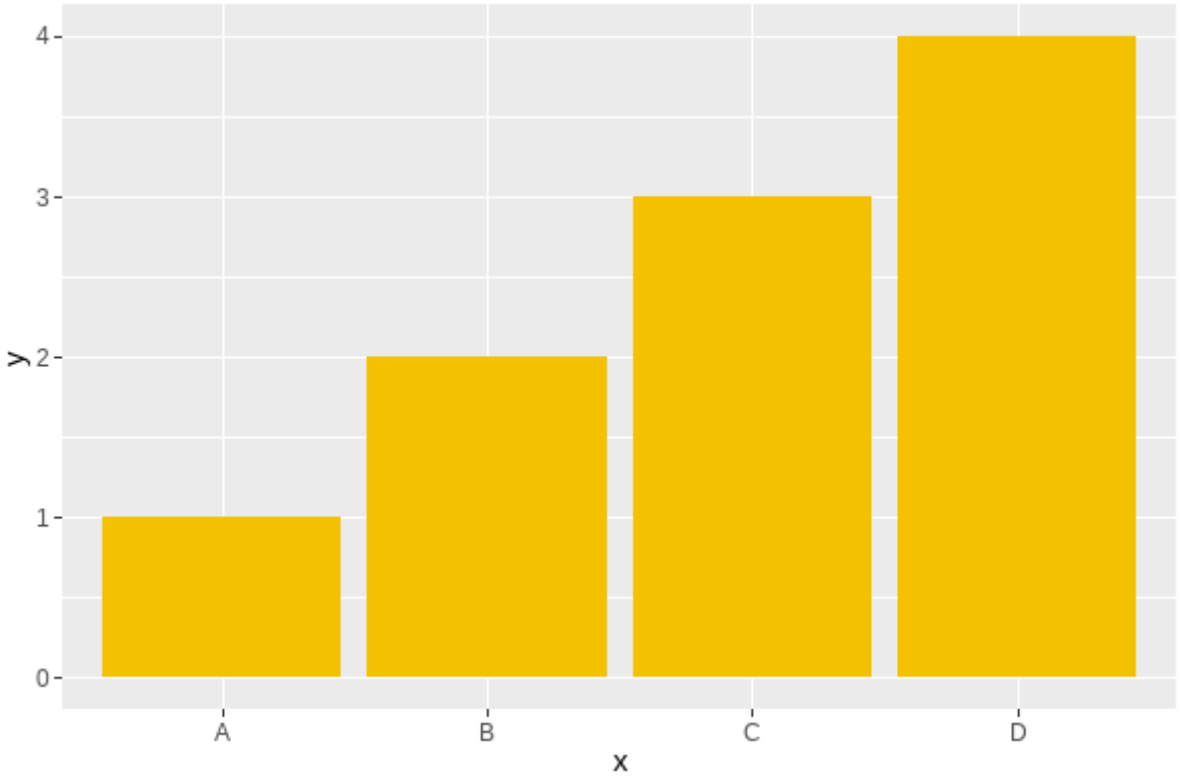


Figure 24: 노란색으로 칠해진 막대가 있는 막대 차트.

예: {ggplot2}에서 이산 색상 스케일 사용.

정성적(이산) 데이터로 작업할 때 사용할 수 있는 가장 좋은 팔레트는 `"signif_qual"`입니다. 이 팔레트에는 현재 4가지 색상만 포함되어 있습니다.

- 이산 (채우기) 스케일: `scale_fill_rss_d()`

```
ggplot(data = plot_df,
       mapping = aes(x = x, y = y, fill = x)) +
  geom_col() +
  scale_fill_rss_d(palette = "signif_qual")
```

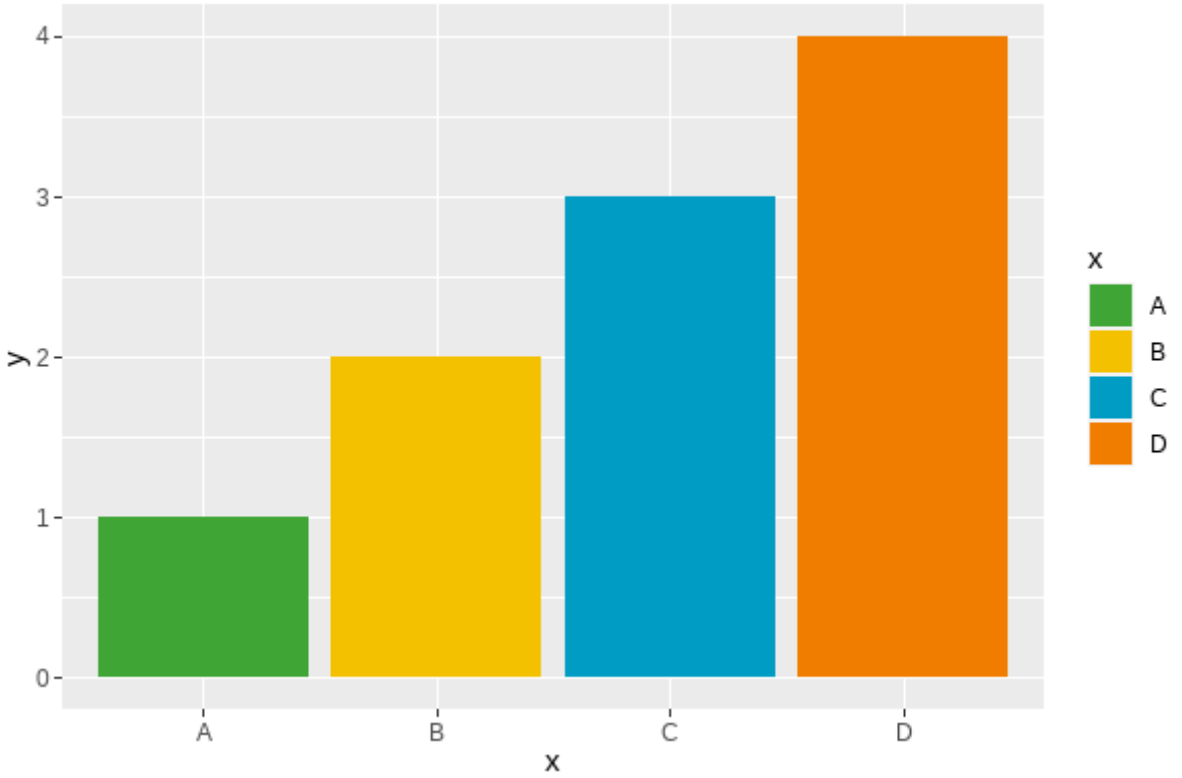


Figure 25: `signif_qual`의 색상이 있는 막대 차트.

- 이산 (색상) 스케일: `scale_colour_rss_d()`

```
ggplot(data = plot_df,
       mapping = aes(x = x, y = y, colour = x)) +
  geom_point(size = 4) +
  scale_colour_rss_d(palette = "signif_qual")
```

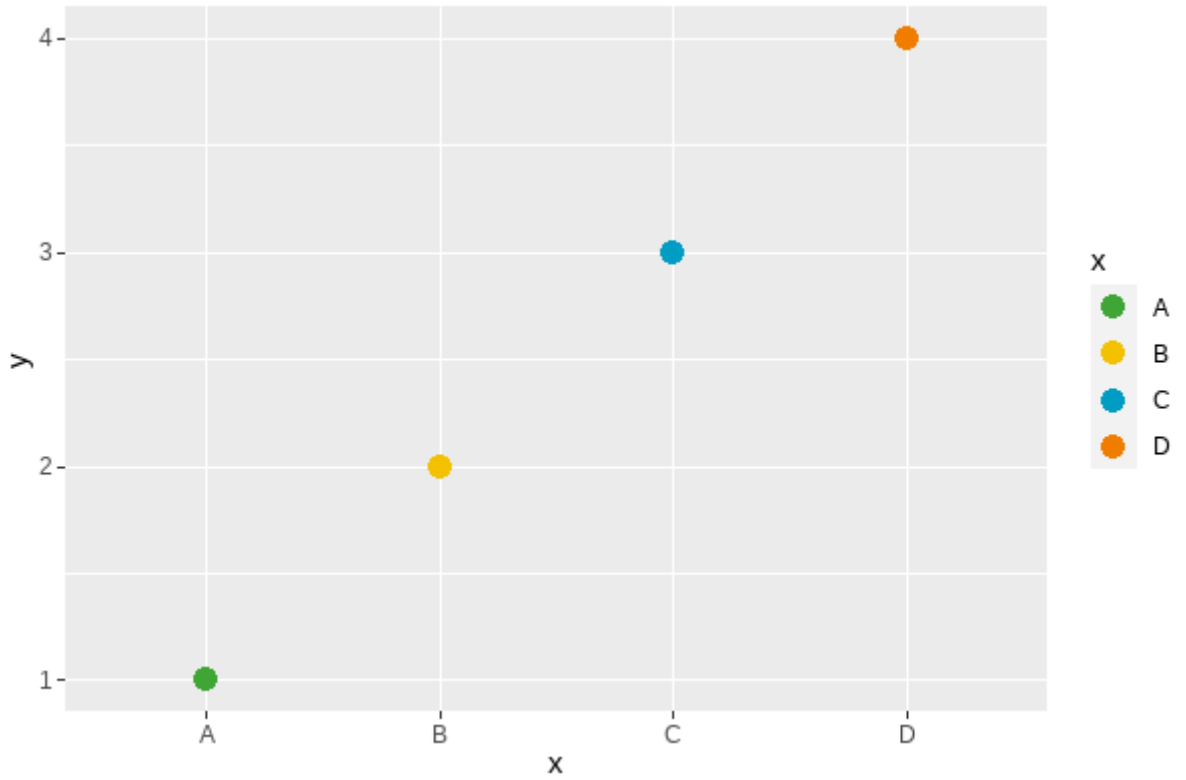


Figure 26: `signif_qual`의 색상이 있는 산점도.

예: {ggplot2}에서 연속 색상 스케일 사용.

연속 색상 스케일은 순차적이거나 발산적일 수 있습니다. 순차적(연속) 데이터로 작업할 때 사용할 수 있는 가장 좋은 팔레트는 `"signif_seq"`입니다.

- 연속 (채우기) 스케일: `scale_fill_rss_c()`

```
ggplot(data = plot_df,
       mapping = aes(x = x, y = y, fill = y)) +
  geom_col() +
  scale_fill_rss_c(palette = "signif_seq")
```

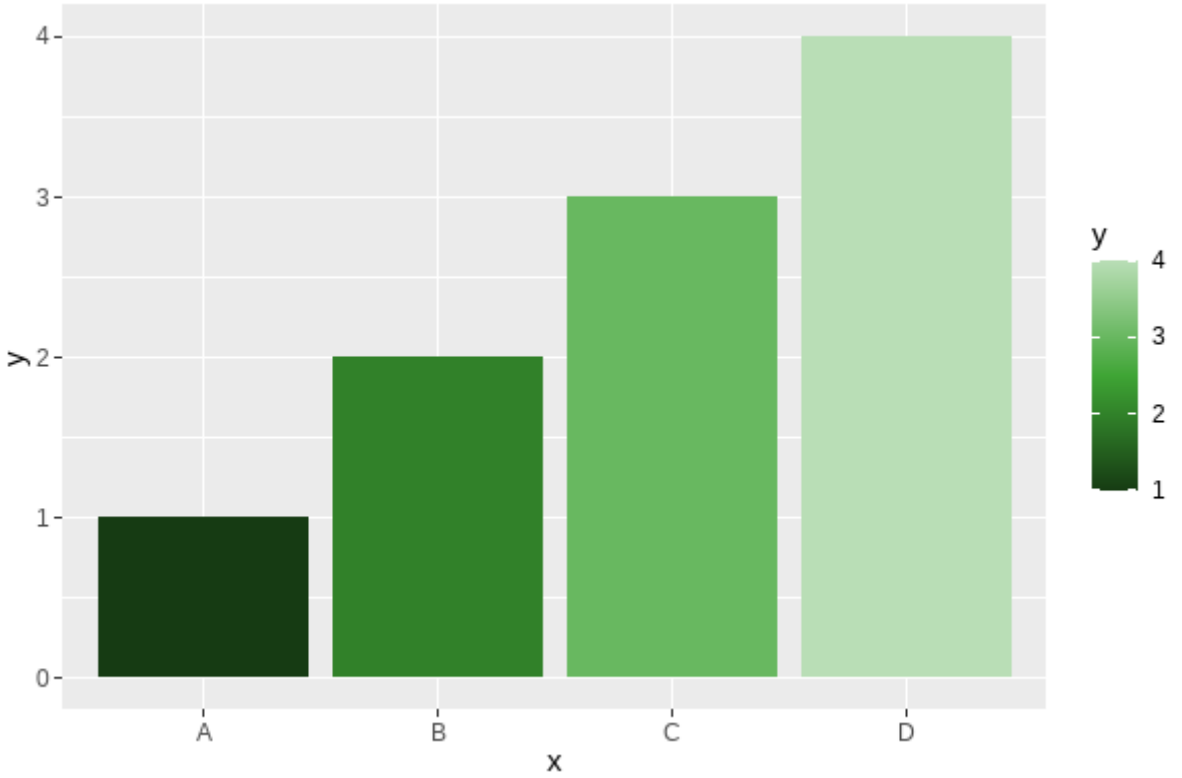


Figure 27: 순차적 녹색 색상 팔레트를 보여주는 막대 차트.

- 연속 (색상) 스케일: `scale_colour_rss_c()`

```
ggplot(data = plot_df,
       mapping = aes(x = x, y = y, colour = y)) +
  geom_point(size = 4) +
  scale_colour_rss_c(palette = "signif_seq")
```

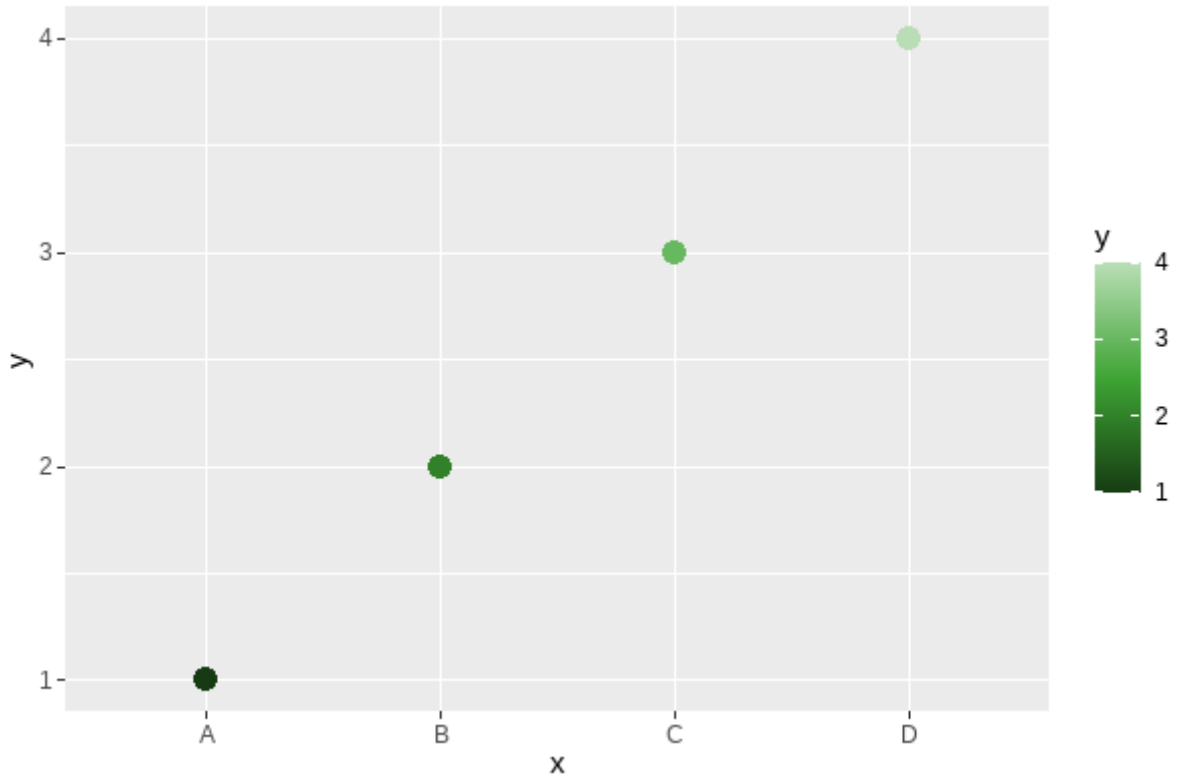


Figure 28: 순차적 녹색 색상 팔레트를 보여주는 산점도.

발산적(연속) 데이터로 작업할 때 사용할 수 있는 가장 좋은 팔레트는 `"signif_div"`입니다.

- 연속 (채우기) 스케일: `scale_fill_rss_c()`

```
ggplot(data = plot_df,
       mapping = aes(x = x, y = y, fill = y)) +
  geom_col() +
  scale_fill_rss_c(palette = "signif_div")
```

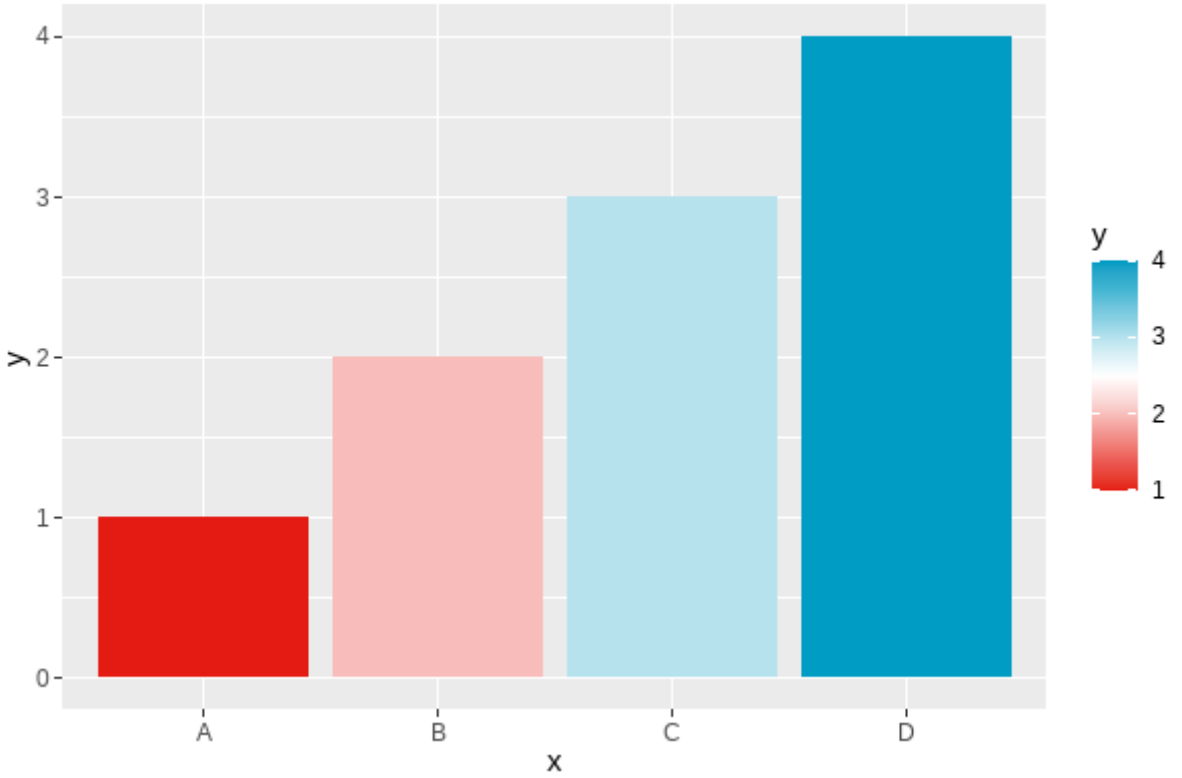


Figure 29: 발산적 빨간색에서 파란색 팔레트를 보여주는 막대 차트.

발산적 스케일을 다른 값을 중심으로 맞추려면 {RSSthemes}에서 미리 정의된 색상을 {ggplot2}의 `scale_fill_gradient2()`에 전달할 수도 있습니다:

```
ggplot(data = plot_df,
       mapping = aes(x = x, y = y, fill = y)) +
  geom_col() +
  scale_fill_gradient2(low = signif_red, high = signif_blue, midpoint = 2)
```

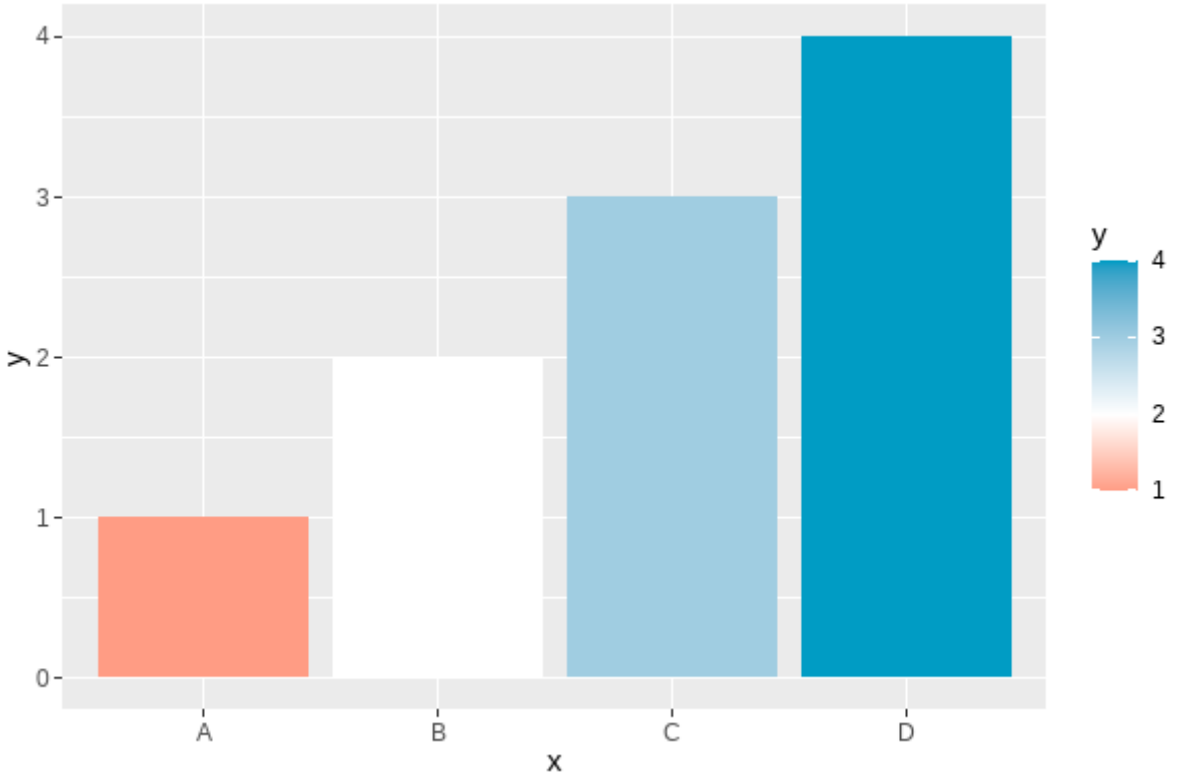


Figure 30: 2를 중심으로 하는 발산적 빨간색에서 파란색 팔레트를 보여주는 막대 차트.

예: {ggplot2}에서 테마 변경.

{ggplot2} 내에서 테마를 사용하면 플롯의 비 데이터 요소 모양을 제어할 수 있습니다. 기본 테마는 `theme_grey()`로 배경이 더 어둡습니다. `theme_minimal()` 또는 `theme_bw()`로 만든 것과 같은 흰색 또는 투명 배경을 사용하는 것이 좋습니다.

또한 {RSSthemes}의 `theme_significance()`를 사용하여 플롯 글꼴을 Significance 매거진에서 사용하는 글꼴 중 하나로 추가 설정할 수 있습니다. 글꼴이 올바르게 로드되도록 `library(RSSthemes)`를 이미 실행했는지 확인하십시오.

```
ggplot(data = plot_df,
       mapping = aes(x = x, y = y, fill = x)) +
  geom_col() +
  labs(title = "  Significance  ",
       subtitle = "                .",
       caption = " :       ") +
  scale_fill_rss_d(palette = "signif_qual") +
  theme_significance()
```

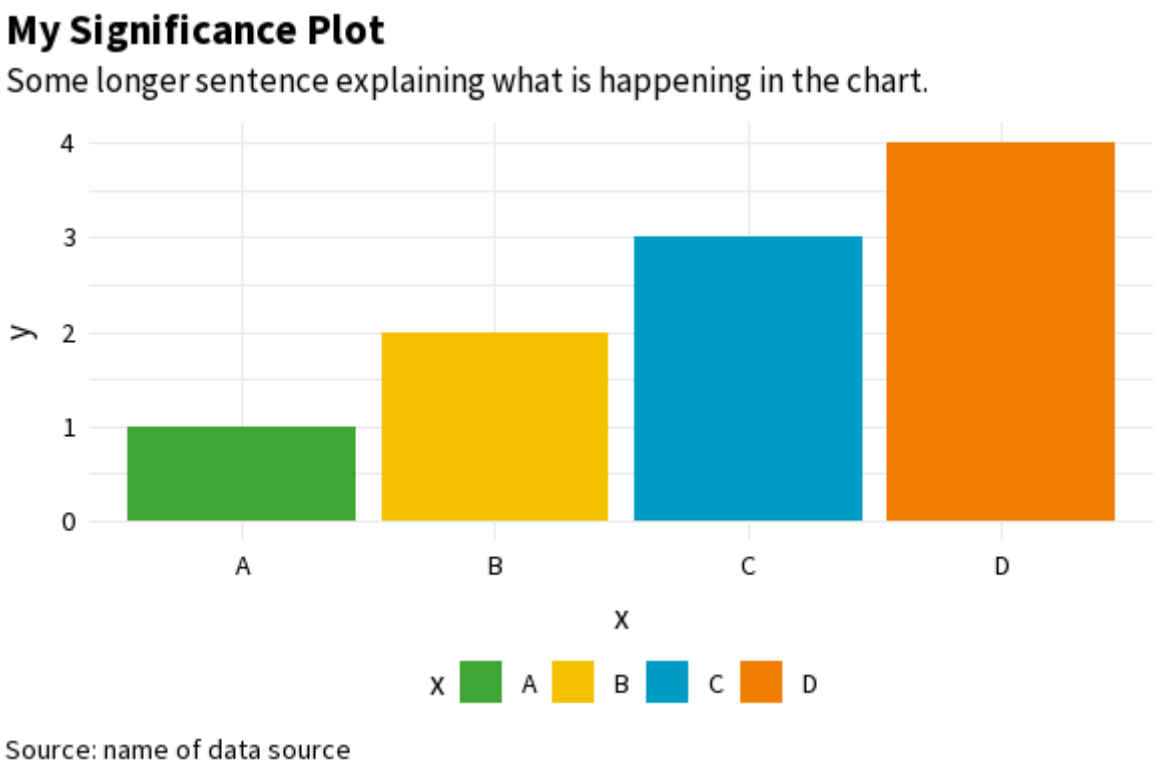


Figure 31: `theme_significance()`로 스타일이 지정된 막대 차트.

{RSSthemes} 패키지에서 버그를 발견하거나 예상대로 작동하지 않는 것이 있으면 GitHub 이슈를 제출해 주십시오.

R에서 이미지 내보내기

R에서 이미지를 내보내고 저장하는 방법에는 여러 가지가 있습니다. RStudio의 Plots 창에서 Export 버튼을 사용해도 출판 품질 그래픽으로 충분한 고해상도 이미지가 생성되지 않는 경우가 많습니다. 인쇄물로 출판되는 이미지의 최소 이미지 해상도는 300 dpi입니다. {ggplot2}의 `ggsave()`를 사용하는 경우 300 dpi가 기본 해상도입니다. 크기를 조정하기 쉽도록 PDF 또는 EPS 파일 형식으로 이미지를 저장하는 것이 좋습니다.

다양한 RSS 간행물에 대한 특정 이미지 크기에 대한 추가 정보는 출판 사양 섹션에 나와 있습니다.

예를 들어 Significance 매거진의 Features 섹션에 대한 플롯을 만들고 플롯이 3개 열 중 2개 열에 걸치도록 하려는 경우를 가정해 보겠습니다. 아래 표에서 이미지 너비는 124mm여야 합니다. `pdf()` 함수를 사용하여 이미지를 저장하려면 너비와 높이가 인치 단위여야 합니다(124mm ~ 4.88인치). 2:1 종횡비를 원하는 경우 높이를 너비의 절반으로 만듭니다:

```
pdf(file = "significance-feature-plot.pdf", #
    width = 4.88,                           #
    height = 4.88 / 2                       #
    )
plot(1:4, 1:4)
dev.off()
```

Python

Python은 범용 프로그래밍 언어로, 데이터 분석 및 시각화 기능을 제공하는 라이브러리를 사용할 수 있습니다.

RSS 색상 팔레트 구현을 위한 진행 중인 Python 패키지는 github.com/nrennie/RSSpythemes에서 찾을 수 있습니다. 이 Python 패키지 개발에 기여하고 싶다면 GitHub에서 이슈를 제기하거나 풀 리퀘스트를 생성해 주십시오.

Matplotlib

Matplotlib은 정적, 애니메이션 및 대화형 데이터 시각화를 생성하기 위한 Python 라이브러리입니다.

예: matplotlib에서 막대 차트 색상 변경.

`color` 인수를 사용하여 matplotlib에서 차트 요소의 색상을 변경할 수 있습니다:

```
import matplotlib.pyplot as plt
# generate data
x_vals = ['A', 'B', 'C', 'D']
y_vals = [1, 2, 3, 4]
# create barchart
plt.bar(x_vals, y_vals, color = "#009cc4")
plt.show()
```

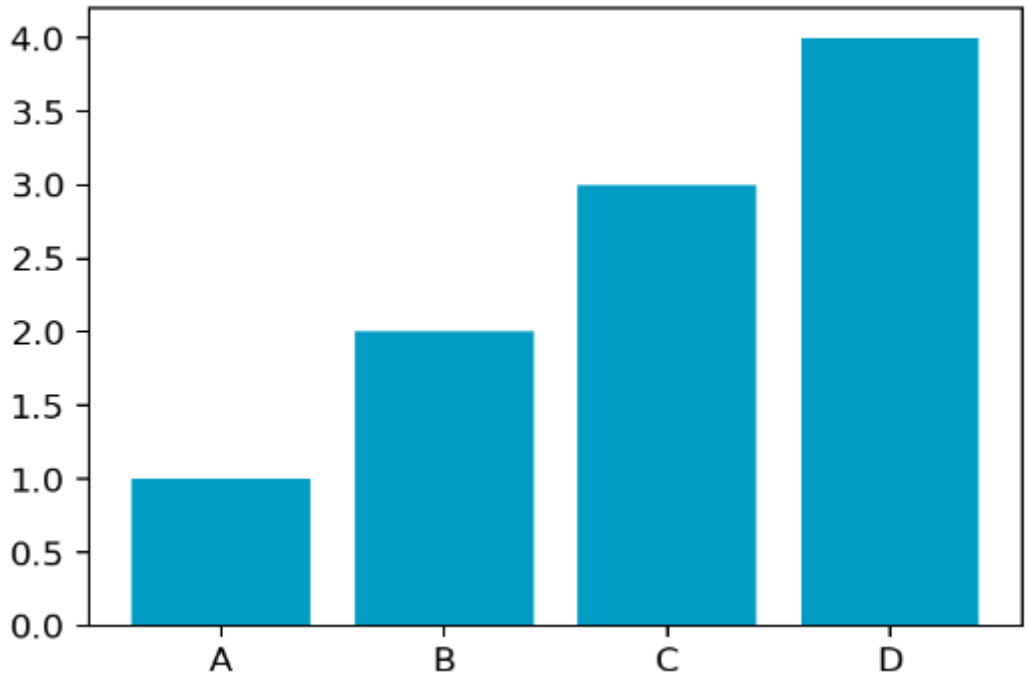


Figure 32: Significance 파란색으로 칠해진 막대가 있는 막대 차트.

플롯의 색상이 데이터의 값을 기반으로 하는 경우 색상 목록을 제공하여 사용되는 색상을 변경할 수도 있습니다:

```
# define colour palette
signif_qual = ['#3fa535', '#f4c100', '#009cc4', '#f07d00']
# create barchart
plt.bar(x_vals, y_vals, label = x_vals, color = signif_qual)
plt.show()
```

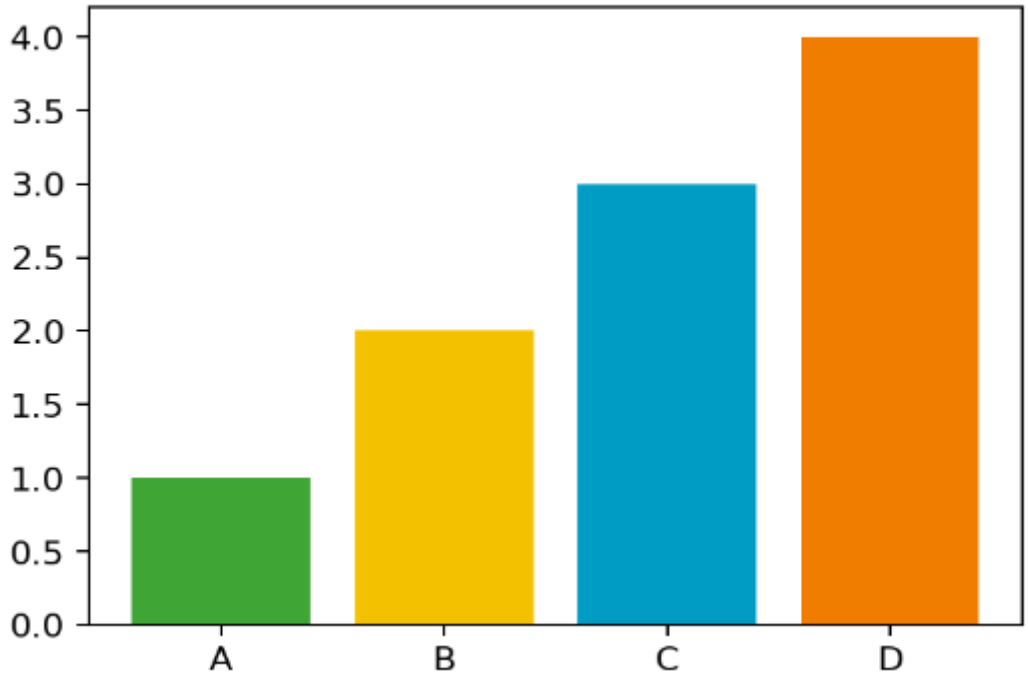


Figure 33: `signif_qual` 팔레트의 색상을 보여주는 막대 차트.

예: matplotlib에서 글꼴 패밀리 변경.

`rcParams`를 사용하여 플롯의 모든 요소에 사용되는 글꼴을 변경할 수 있습니다. 사용자 지정 글꼴 패밀리를 설정할 때 좋은 방법은 일반 글꼴 패밀리(예: sans serif)를 백업으로 추가하는 것입니다. 시스템에 미리 설치되지 않은 글꼴을 사용하는 경우 `font_manager`를 사용하여 글꼴을 로드할 수 있습니다:

```
from matplotlib import font_manager
font_manager.fontManager.addfont("SourceSans3-Regular.ttf")
```

개별 요소에 대해 `fontdic`을 사용하여 다른 글꼴 패밀리, 두께 또는 크기를 지정할 수 있습니다.

```
# define fonts
from matplotlib import rcParams
rcParams['font.family'] = ['Source Sans 3', 'sans-serif']

# create barchart
fig, ax = plt.subplots(1, 1)
plt.bar(x_vals, y_vals, color = signif_qual, label = x_vals)
```

```
plt.title('  Significance  ', fontdict = {'fontsize':14}, loc = 'left')
# add grid lines lines
ax.set_axisbelow(True)
ax.xaxis.grid(color = 'lightgrey')
ax.yaxis.grid(color = 'lightgrey')
# add legend below plot
ax.legend(ncol = 4, loc = 'lower center',
          bbox_to_anchor = (0.5, -0.15), frameon = False)
plt.show()
```

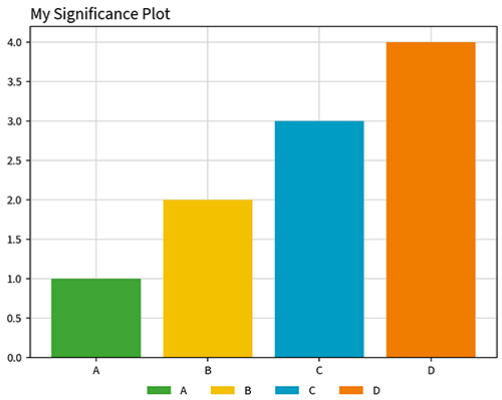


Figure 34: Source Sans 글꼴 사용을 보여주는 막대 차트.

Julia

Julia는 고성능에 초점을 맞추면서도 쉬운 구문을 제공하는 범용 프로그래밍 언어입니다. Julia에는 데이터 분석 및 시각화 기능을 제공하는 라이브러리가 있습니다.

Makie

Makie는 Julia 프로그래밍 언어를 위한 고수준 플로팅 라이브러리입니다.

예: Makie에서 막대 차트 색상 변경.

Makie에서는 `color` 인수를 사용하여 차트 요소의 색상을 변경하고 `axis` 내부의 `xticks` 인수를 사용하여 사용자 지정 눈금 레이블을 변경할 수 있습니다:

```
using CairoMakie
# generate data
x_vals = [1, 2, 3, 4]
y_vals = [1, 2, 3, 4]
# create barchart
barplot(x_vals, y_vals, color="#009cc4", axis=(; xticks=(1:4, ["A", "B", "C", "D"])))
```

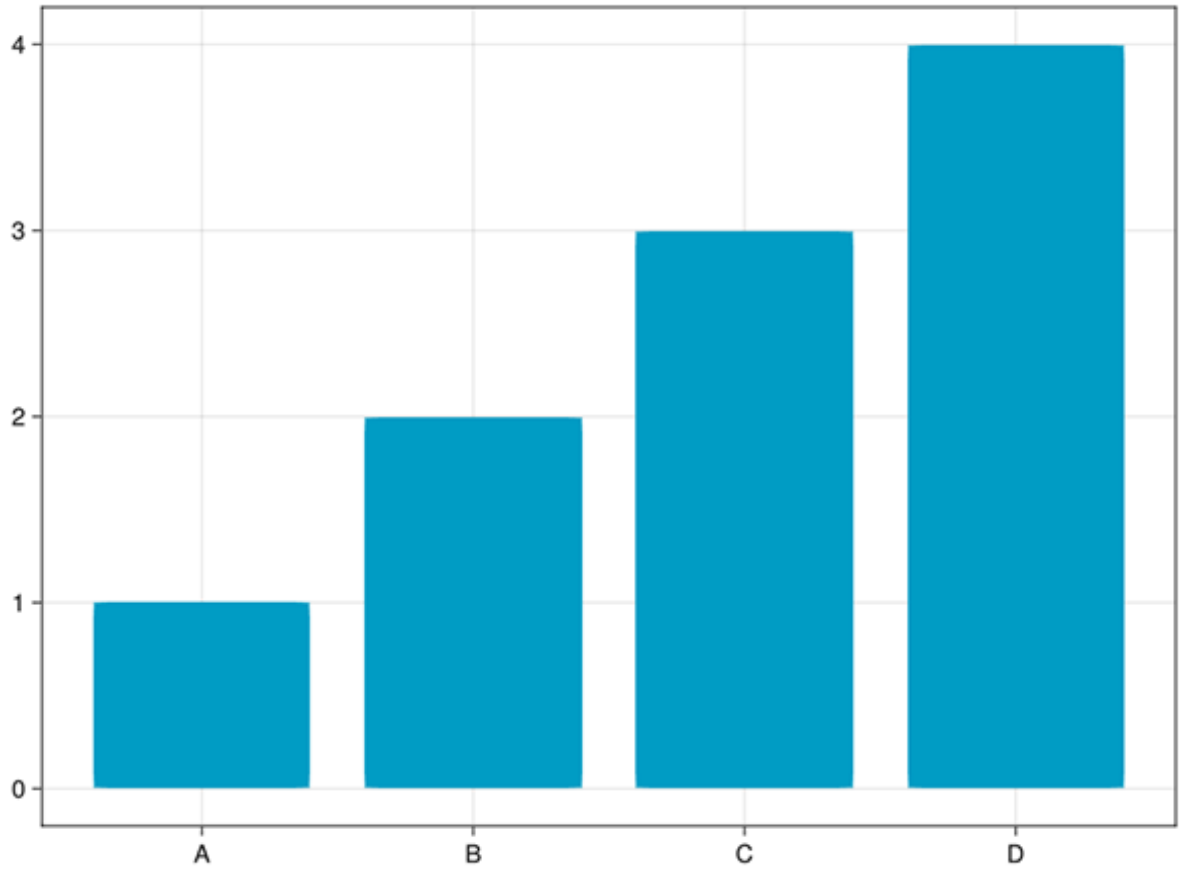


Figure 35: Significance 파란색으로 칠해진 막대가 있는 막대 차트.

플롯의 색상이 데이터의 값을 기반으로 하는 경우 색상 목록을 제공하여 사용되는 색상을 변경할 수도 있습니다:

```
# define colour palette
signif_qual = ["#3fa535", "#f4c100", "#009cc4", "#f07d00"]
# create barchart
barplot(x_vals, y_vals, color=signif_qual, axis=(; xticks=(1:4, ["A", "B", "C", "D"])))
```

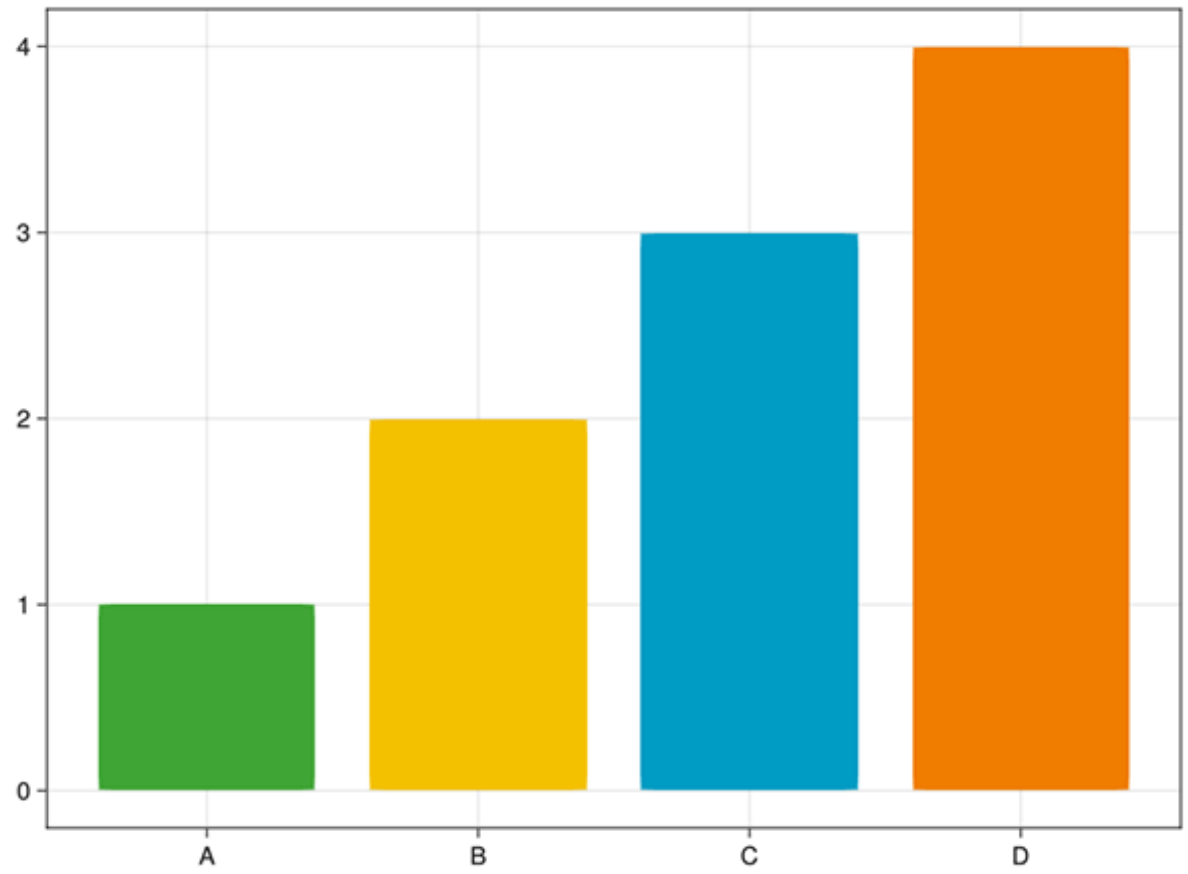


Figure 36: `signif_qual` 팔레트의 색상을 보여주는 막대 차트.

`axis` 인수를 사용하여 사용자 지정 레이블과 제목을 지정할 수 있습니다:

```
# define labels and title
title = "  Significance  "
subtitle = "               ."
xlabel = "X   "
ylabel = "Y   "

# create barchart
barplot(x_vals,
        y_vals;
        color=signif_qual,
        axis=(;
            xticks=(1:4, ["A", "B", "C", "D"]),
            title=title,
            subtitle=subtitle,
            titlealign=:left,
            xlabel=xlabel,
            ylabel=ylabel,
        ),
)
```

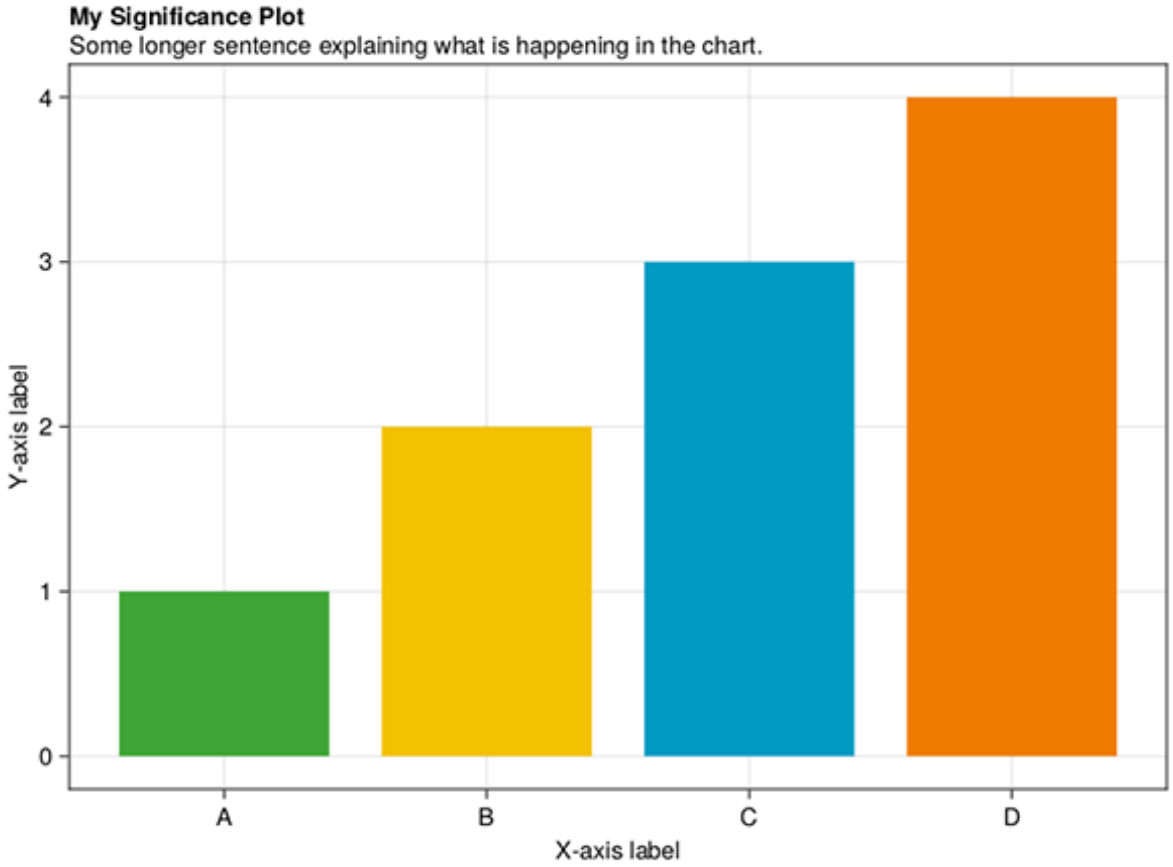


Figure 37: 사용자 지정 레이블 및 제목 사용을 보여주는 막대 차트.

AlgebraOfGraphics

AlgebraOfGraphics는 플로팅 백엔드로 Makie를 사용하는 Julia 프로그래밍 언어를 위한 고수준 플로팅 라이브러리입니다. 그래픽 문법을 따르며 R의 `ggplot2`와 유사합니다.

예: AlgebraOfGraphics에서 막대 차트 색상 변경.

Makie에서는 `color` 인수를 사용하여 차트 요소의 색상을 변경하고 `axis` 내부의 `xticks` 인수를 사용하여 사용자 지정 눈금 레이블을 변경할 수 있습니다:

```
using AlgebraOfGraphics
using CairoMakie
# generate data
x_vals = [1, 2, 3, 4]
y_vals = [1, 2, 3, 4]
# create barchart
plt = data((; x_vals, y_vals)) * mapping(:x_vals, :y_vals) * visual(BarPlot; color="#009cc4"
draw(plt; axis=(; xticks=(1:4, ["A", "B", "C", "D"])))
```

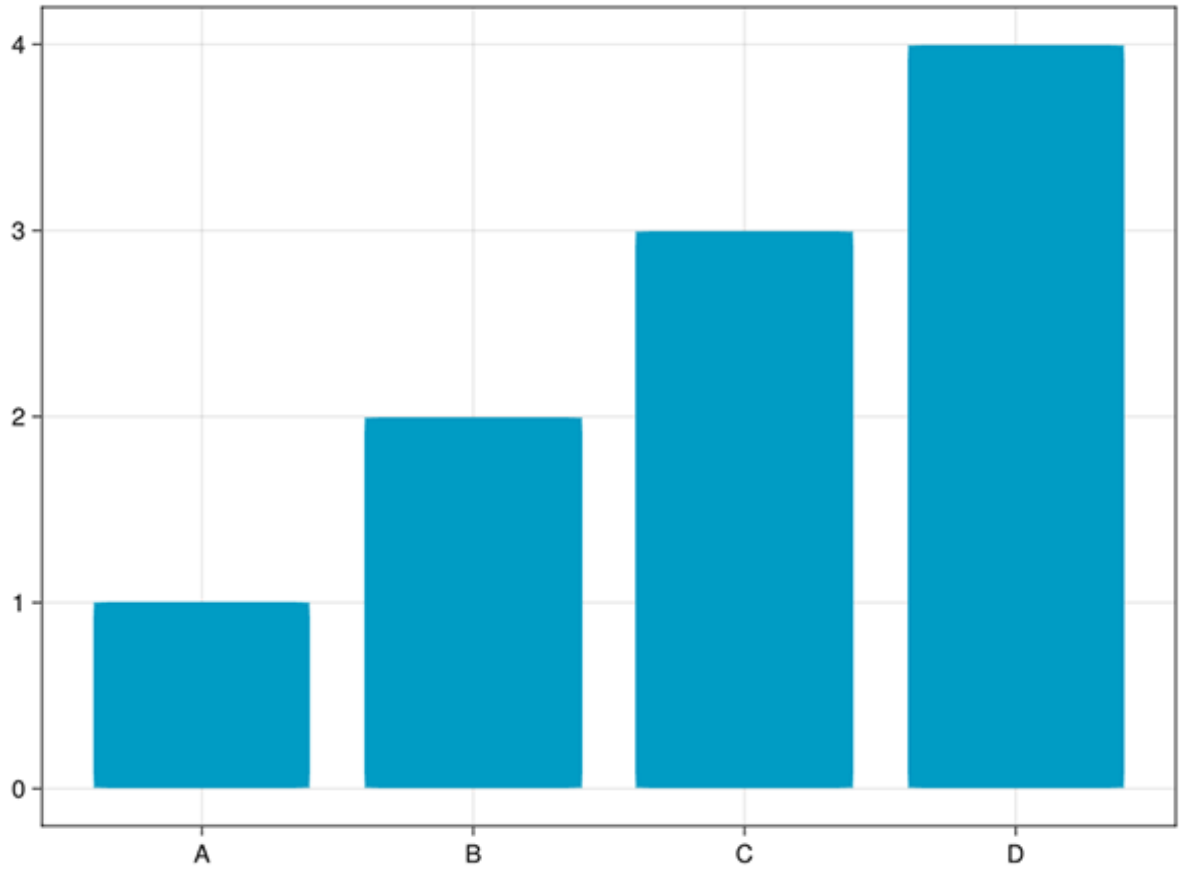


Figure 38: Significance 파란색으로 칠해진 막대가 있는 막대 차트.

플롯의 색상이 데이터의 값을 기반으로 하는 경우 색상 목록을 제공하여 사용되는 색상을 변경할 수도 있습니다:

```
# define colour palette
signif_qual = ["#3fa535", "#f4c100", "#009cc4", "#f07d00"]
# create barchart
plt = data((; x_vals, y_vals)) * mapping(:x_vals, :y_vals) * visual(BarPlot; color=signif_qu
draw(plt; axis=(; xticks=(1:4, ["A", "B", "C", "D"])))
```

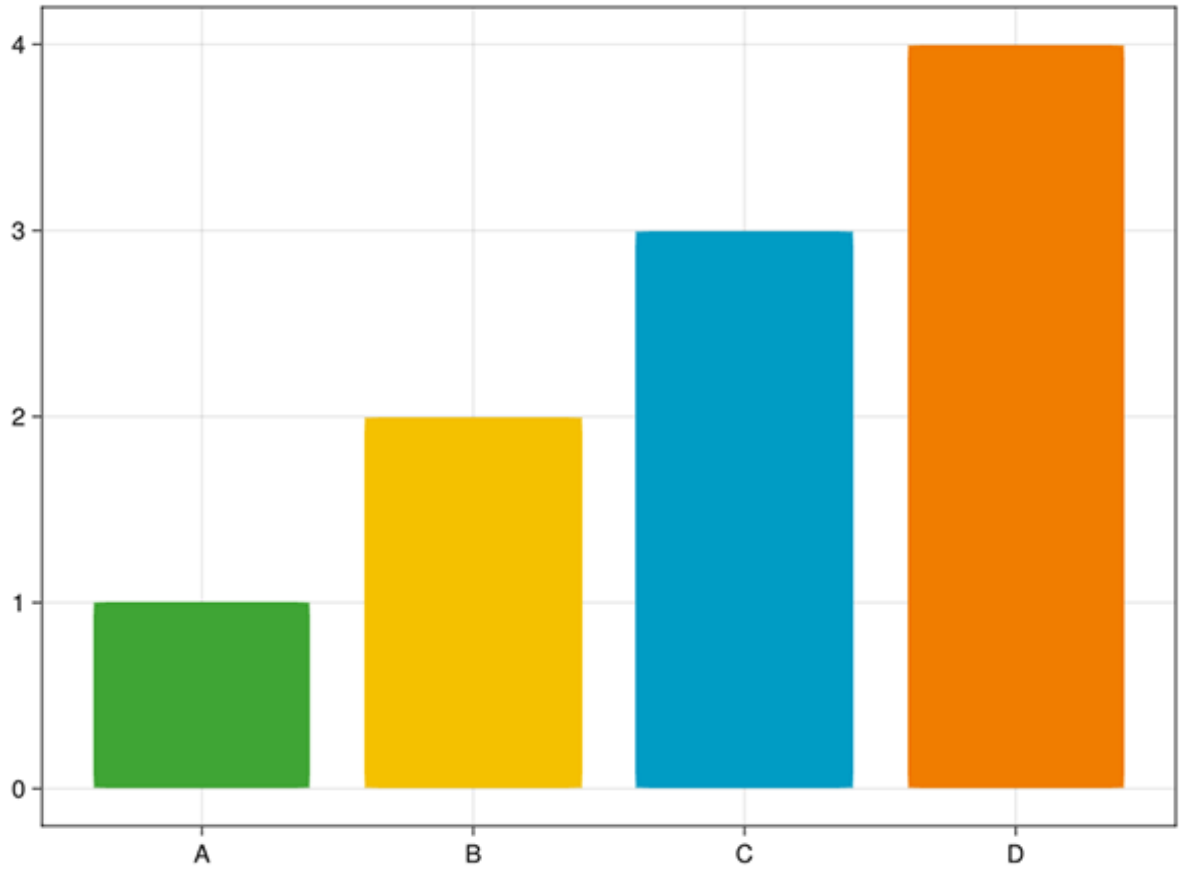


Figure 39: `signif_qual` 팔레트의 색상을 보여주는 막대 차트.

`axis` 인수를 사용하여 사용자 지정 레이블과 제목을 지정할 수 있습니다:

```
# define labels and title
title = "  Significance  "
subtitle = "               ."
xlabel = "X   "
ylabel = "Y   "

# create barchart
draw(plt;
     axis=(;
         xticks=(1:4, ["A", "B", "C", "D"]),
         title=title,
         subtitle=subtitle,
         titlealign=:left,
         xlabel=xlabel,
         ylabel=ylabel,
     ),
)
```

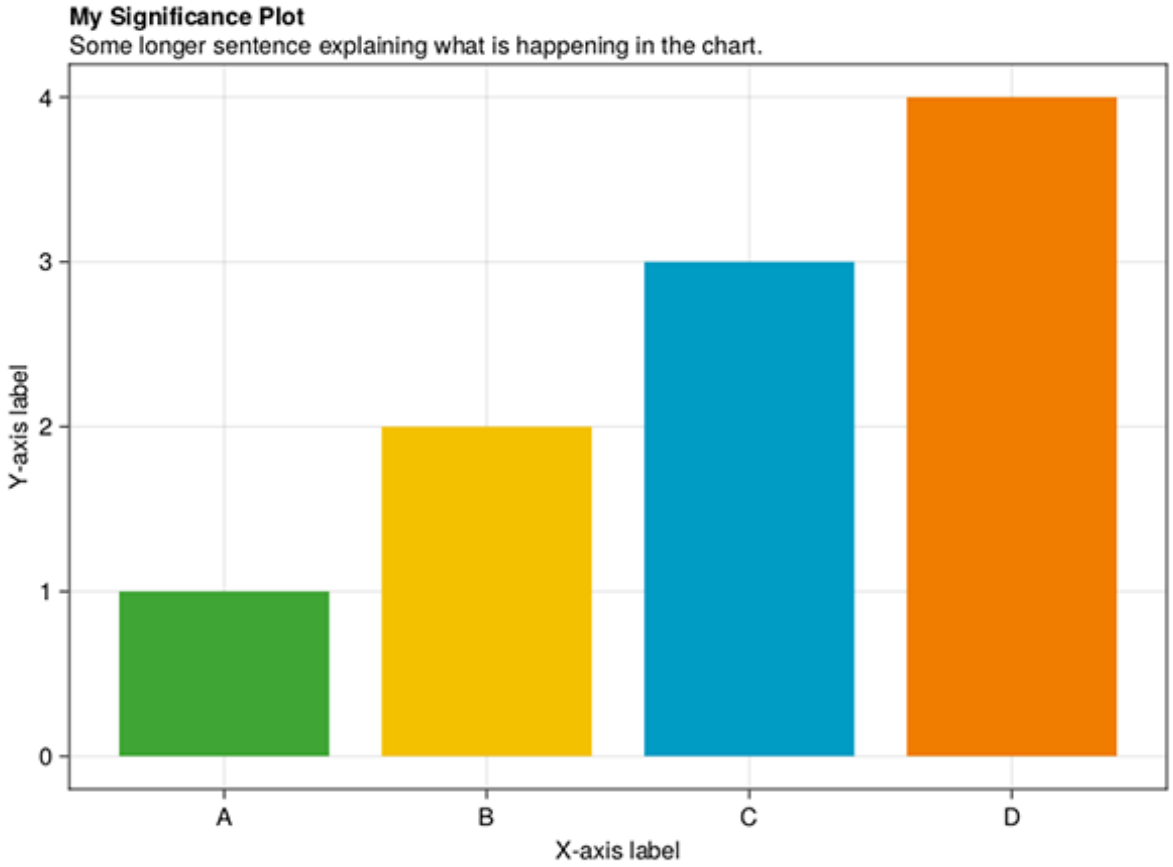


Figure 40: 사용자 지정 레이블 및 제목 사용을 보여주는 막대 차트.

7.2 출판 사양

다음 정보는 RSS 출판 요구 사항을 충족하는 그래프 및 차트를 디자인하는 데 사용해야 합니다. 세부 정보에는 페이지 크기 및 열 너비, 글꼴 유형 및 크기, 이미지 해상도 및 파일 형식이 포함됩니다.

Significance Magazine

페이지 크기	(W) 212.55 mm x (H) 263.65 mm
텍스트 영역	(W) 188 mm x (H) 212 mm
이미지 해상도	300 dpi (인쇄 품질)
권장 이미지 파일 형식	jpeg, png

Notebook 섹션

4열 레이아웃을 사용합니다.

1x 열 너비	45 mm
2x 열 너비	93 mm
3x 열 너비	140 mm
4x 열 너비	188 mm

본문 글꼴	Meta Serif OT, Book
글꼴 크기	8.5 pt
섹션 색상	빨강: (`RSSthemes` 패키지에서 `signif_red` 사용) • CMYK = 0, 96, 98, 1 • RGB = 228, 27, 18 • Hex code = #e41b12

Features 섹션

3열 레이아웃을 사용합니다.

1x 열 너비	60 mm
2x 열 너비	124 mm
3x 열 너비	188 mm
본문 글꼴	Source Sans Pro, Regular
글꼴 크기	9 pt
섹션 색상	녹색: (`RSSthemes` 패키지에서 `signif_green` 사용) • CMYK = 75, 5, 100, 0 • RGB = 63, 165, 53 • Hex code = #3fa535

Profiles / Perspectives / Statscom 섹션

3열 레이아웃을 사용합니다.

1x 열 너비	60 mm
2x 열 너비	124 mm
3x 열 너비	188 mm
본문 글꼴	Meta Serif OT, Book
글꼴 크기	8.5 pt
섹션 색상:	

Profiles	파랑: (`RSSthemes` 패키지에서 `signif_blue` 사용) • CMYK = 78, 19, 15, 1 • RGB = 0, 156, 196 • Hex code = #009cc4
Perspectives	노랑: (`RSSthemes` 패키지에서 `signif_yellow` 사용) • CMYK = 5, 24, 95, 1 • RGB = 244, 193, 0 • Hex code = #f4c100
Statscomm	주황: (`RSSthemes` 패키지에서 `signif_orange` 사용) • CMYK = 0, 60, 99, 0 • RGB = 240, 125, 0 • Hex code = #f07d00

Journal of the Royal Statistical Society Series A

단일 열 레이아웃을 사용합니다.

페이지 크기	(W) 189 mm x (H) 246 mm
텍스트 영역	(W) 136 mm x (H) 217 mm
본문 글꼴	Sabon LT Std Roman
글꼴 크기	9.25 pt
이미지 해상도	300 dpi (인쇄 품질)
권장 이미지 파일 형식	jpeg, png

8 참고 문헌

8.1 가이드에서 참조된 텍스트

Battle-Baptiste, W., and B. Rusert. 2018. W.e.b. Du Bois's Data Portraits: Visualizing Black America: The Color Line at the Turn of the Twentieth Century. The W.E.B. Du Bois Center at the University of Massachusetts.

Becker, Richard A., and John M. Chambers. 1984. S: An Interactive Environment for Data Analysis and Graphics. Pacific Grove, CA: Wadsworth & Brooks/Cole.
Becker, Richard A., and William S. Cleveland. 1996. S-PLUS Trellis Graphics User's Manual. Seattle, WA: MathSoft.
Beecham, Roger, Jason Dykes, Layik Hama, and Nik Lomax. 2021. "On the Use of 'Glyphmaps' for Analysing the Scale and Temporal Spread of COVID-19 Reported Cases." ISPRS International Journal of Geo-Information 10 (4). https://doi.org/10.3390/ijgi10040213.
Cesal, Amy. 2020. "Writing Alt Text for Data Visualization." Nightingale. 2020. https://medium.com/nightingale/writing-alt-text-for-data-visualization-2a218ef43f81.
"Chart Titles and Text." n.d. Office for National Statistics. Accessed July 10, 2023. https://style.ons.gov.uk/data-visualisation/titles-and-text/annotation-and-footnotes/.
Cleveland, William S. 1993. Visualizing Data. Summit, NJ: Hobart Press.
———. 1994. The Elements of Graphing Data. Summit, NJ: Hobart Press.
"Coblis — Color Blindness Simulator." n.d. Colblindor. Accessed July 10, 2023. https://www.color-blindness.com/coblis-color-blindness-simulator/.
Corbett, J. 2001. "Charles Joseph Minard, Mapping Napoleon's March, 1861." CSISS Class 2001. 2001. https://escholarship.org/uc/item/4qj8h064.
D'Ignazio, C., and L. F. Klein. 2020. Data Feminism. MIT Press. https://mitpress.mit.edu/9780262547185/data-feminism/.
"Documentation." n.d. Statistical Analysis System. Accessed June 19, 2023. https://support.sas.com/en/documentation.html.
Du Bois, W. E. B. 1900. The Exhibit of American Negroes. Paris.
Few, Stephen. 2004. Show Me the Numbers. Burlingame, CA: Analytics Press.
Friendly, M. 2018. "A Very Brief History of Visualization: Visions, Stories and Pictures." Chicago, IL. 2018. http://datavis.ca/papers/CHF-2x2.pdf.
———. 2022. "Remembrances of Things EDA." 2022. https://www.researchgate.net/publication/361191335_Rememberances_of_Things_EDA.
Friendly, M., and D. Denis. 2005. "The Early Origins and Development of the Scatterplot." J. Hist. Behav. Sci. 41 (2): 103–30. https://doi.org/10.1002/jhbs.20078.
Garland, Kevin. 1994. Mr Beck's Underground Map. Capital Transport.
Green, Nathan. 2023. "Why your data viz needs alt text." Significance 20 (1): 38–39. https://doi.org/10.1093/jrssig/qmad011.
Hedley, Alison. 2020. "Florence Nightingale and Victorian Data Visualisation." Significance 17 (2): 26–30. https://doi.org/10.1111/1740-9713.01376.
Kent, AJ. 2021. "When Topology Trumped Topography: Celebrating 90 Years of Beck's Underground Map." The Cartographic Journal 58 (1): 1–12. https://doi.org/10.1080/00087041.2021.1953765.
Krause, Andreas. 2013. "Concepts and Principles of Clinical Data Graphics." In A Picture Is Worth a Thousand Tables: Graphics in Life Sciences, 3–21. Springer. https://doi.org/10.1007/978-1-4614-5329-1_1.
Krause, Andreas, and Michael O'Connell, eds. 2013. A Picture Is Worth a Thousand Tables: Graphics in Life Sciences. Springer. https://doi.org/10.1007/978-1-4614-5329-1.
Muth, Lisa. 2018. "An Alternative to Pink & Blue: Colors for Gender Data." Datawrapper. 2018. https://blog.datawrapper.de/gendercolor/.
Nightingale, Florence. 1859. "A Contribution to the Sanitary History of the British Army During the Late War with Russia." London, UK: Harrison; Sons. 1859. https://iiif.lib.harvard.edu/manifests/view/drs:7420433$24b.
Norman, Donald A. 1990. The Design of Everyday Things. New York, NY: Currency Doubleday.

R Core Team. 2021. R: A Language and Environment for Statistical Computing. Vienna, Austria: R Foundation for Statistical Computing. https://www.R-project.org/.
Rello, Luz, and Ricardo Baeza-Yates. 2016. "The Effect of Font Type on Screen Readability by People with Dyslexia." ACM Trans. Access. Comput. 8 (4). https://doi.org/10.1145/2897736.
Robbins, Naomi B. 2006. Creating More Effective Graphs. Hoboken, NJ: Wiley.
Robinson, A. H. 1967. "The Thematic Maps of Charles Joseph Minard." Imago Mundi 21: 95–108.
Sarkar, Deepan. 2008. Lattice: Multivariate Data Visualization with r. New York, NY: Springer.
Schwabish, Jonathan. 2021. Better Data Visualizations: A Guide for Scholars, Researchers, and Wonks. Columbia University Press. http://www.jstor.org/stable/10.7312/schw19310.
Setlur, V., and B. Cogley. 2022. Functional Aesthetics for Data Visualization. Wiley. https://www.functionalaestheticsbook.com/.
Snow, John. 1854. "Mode of Communication of Cholera." Piccadilly (London), UK: John Churchill. 1854. https://archive.org/details/b28985266/page/52/mode/2up?view=theater.
"Styling Base r Graphics." 2018. Jumping Rivers. 2018. https://www.jumpingrivers.com/blog/styling-base-r-graphics/.
Tennekes, Martijn, and Marco J. H. Puts. 2023. "cols4all: a Color Palette Analysis Tool." In EuroVis 2023 - Short Papers, edited by Thomas Hoellt, Wolfgang Aigner, and Bei Wang. The Eurographics Association. https://doi.org/10.2312/evs.20231040.
Tol, Paul. 2021. "Introduction to Colour Schemes." 2021. https://personal.sron.nl/~pault/.
Tufte, Edward R. 1990. Envisioning Information. Graphics Press.
———. 2001. The Visual Display of Quantitative Information. 2nd ed. Cheshire, CT: Graphics Press.
———. 2004. Visual Explanations: Images and Quantities, Evidence and Narrative. Cheshire, CT: Graphics Press.
———. 2006. Beautiful Evidence. Cheshire, CT: Graphics Press.
Tukey, John W. 1977. Exploratory Data Analysis. Reading, MA: Addison-Wesley.
van Weert, Julia C. M., Monique C. Alblas, Liset van Dijk, and Jesse Jansen. 2021. "Preference for and Understanding of Graphs Presenting Health Risk Information. The Role of Age, Health Literacy, Numeracy and Graph Literacy." Patient Education and Counseling 104 (1): 109–17. https://doi.org/10.1016/j.pec.2020.06.031.
Wery, J. J., and J. A. Diliberto. 2017. "The Effect of a Specialized Dyslexia Font, OpenDyslexic, on Reading Rate and Accuracy." Ann. Of Dyslexia. 67: 114–27. https://doi.org/10.1007/s11881-016-0127-1.
Wickham, Hadley. 2011. "Ggplot2." Wiley Interdisciplinary Reviews: Computational Statistics 3: 180–85. https://onlinelibrary.wiley.com/doi/10.1002/wics.147.
———. 2016a. Ggplot2: Create Elegant Data Visualisations Using the Grammar of Graphics. Vol. 2. Use r! Springer International Publishing. https://doi.org/10.1007/978-3-319-24277-4.
———. 2016b. Ggplot2: Elegant Graphics for Data Analysis. Springer-Verlag New York. https://ggplot2.tidyverse.org.
Wikipedia contributors. 2023. "1854 Broad Street Cholera Outbreak." 2023. https://en.wikipedia.org/wiki/1854_Broad_Street_cholera_outbreak.
Wilkinson, Leland. 2005. The Grammar of Graphics. Statistics and Computing. New York: Springer-Verlag. https://doi.org/10.1007/0-387-28695-0.
Williams, T. A., David R. Anderson, and Dennis J. Sweeney. 2023. "Statistics." Encyclo-

pedia Britannica. https://www.britannica.com/science/statistics.

8.2 추가 읽기 자료

Functional Aesthetics for Data Visualization.

Setlur and Cogley (2022)

추천 이유: 이 책은 연구와 실무를 연결하며, 인지 심리학, 데이터 리터러시, 시각 디자인의 아이디어를 결합합니다. 이 책의 아이디어는 단순한 차트를 넘어 대화형 대시보드 및 시각화를 포함한 복잡한 시각적 디스플레이까지 확장될 수 있습니다.

Better Data Visualizations: A Guide for Scholars, Researchers, and Wonks.

Schwabish (2021)

추천 이유: 이 책은 보여주고자 하는 내용에 따라 어떤 유형의 데이터 시각화가 가장 적절한지 안내합니다. 뿐만 아니라 시각적 인지 이론을 사용하여 왜 어떤 차트가 다른 차트보다 읽기 쉬운지 설명합니다.

Data Feminism.

D'Ignazio and Klein (2020)

추천 이유: 우리는 데이터 시각화에 대해 이야기할 때 종종 "스토리텔링"이라는 용어를 듣게 되는데, 여기서 스토리텔러의 역할을 기억해야 합니다. 이 책에서 저자들은 우리가 데이터로 작업할 때 데이터가 수집된 이유와 그것이 무엇을 위해 사용될 수 있는지에 대해서도 생각해야 한다는 점을 포함하여 몇 가지 중요한 점을 지적합니다.

Preference for and understanding of graphs presenting health risk information. The role of age, health literacy, numeracy and graph literacy.

van Weert et al. (2021)

추천 이유: 환자들은 종종 차트와 표 형식의 건강 위험 정보에 대한 데이터를 받습니다. 이 기사는 환자들이 다양한 차트 유형을 통해 자신에게 제시된 정보를 얼마나 잘 이해하는지 조사합니다. 환자들이 선호하는 차트 유형이 항상 가장 잘 이해하는 차트 유형은 아니라는 것을 발견했습니다.

8.3 추가 리소스

교육 과정

데이터 제시 (RSS)

이 온라인 과정은 통계 정보 제시의 기초입니다. 표, 차트, 지도 및 텍스트로 정보를 제시하는 기본 원칙을 설명합니다. 이러한 원칙들은 예시를 통해 설명된 다음 실습을 통해 강화됩니다.

Tableau에서의 데이터 탐색 (RSS)

Tableau는 단순한 데이터 시각화 도구 그 이상입니다. 또한 여러 데이터 소스를 조작하고, 사용자 정의 차트를 만들고, 예측 모델을 구축하고, 플롯을 대화형 대시보드 및 프레젠테이션으로 전환할 수 있는 기능을 제공합니다.

이 가상 과정은 두 번의 오후에 걸쳐 진행됩니다. 데이터를 랭글링(wrangling)하고, 데이터 분석을 수행하고, 결과를 시각적으로 전달하는 과정을 안내합니다. 이 과정을 마치면 자신의 데이터를 조작하여 사용자 정의 차트를 구축할 수 있게 됩니다. Tableau에서 지리적 데이터로 작업하는 방법, 예측 모델을 사용하여 예측하는 방법, 대화형 대시보드 및 스토리를 만들어 작업을 공유하는 방법을 배우게 됩니다.

데이터 탐색 및 기본 통계를 위한 Power BI (RSS)

Power BI는 대화형 보고서 및 대시보드 제작을 위한 표준 도구로 빠르게 자리 잡고 있으며, 이 과정은 데이터 준비부터 시각적 개체 구축 및 공유에 이르기까지 워크플로에 중점을 두고 Power BI에 대한 체계적인 소개를 제공합니다. 의도한 청중을 기반으로 적절한 수준의 기능과 사용성을 달성하는 데 도움이 되도록 디자인 이슈에 중점을 둡니다. 또한 기본 통계 사용에도 특별한 주의를 기울일 것입니다.

ggplot2를 사용하여 R에서 출판 품질 차트 게시하기 (RSS)

이 강사가 주도하는 가상 과정에서는 tidyverse와 ggplot2를 사용하여 R에서 출판 품질의 차트를 재현 가능하게 만드는 방법을 소개합니다.

로우 코드/노 코드 시각화 도구

Datawrapper

https://www.datawrapper.de/

Flourish

https://flourish.studio/

9 저자 소개

9.1 Andreas Krause

Andreas Krause는 통계학 및 컴퓨터 과학 분야에서 박사 학위와 석사 학위를 보유하고 있습니다. 그는 Simulations Plus, Inc.의 임상 약리학 및 약물 계량 서비스(Clinical Pharmacology & Pharmacometric Services) 최고 과학 책임자(Chief Scientific Officer)로서 제약 산업에 모델링 및 시뮬레이션 전문 지식을 제공합니다. 그의 저작물에는 100편 이상의 동료 심사 과학 출판물이 포함됩니다.

Andreas의 과학 활동에는 미국 통계 협회(American Statistical Association)의 통계 컴퓨팅 및 그래픽 섹션 공동 편집자, Computational Statistics의 서평 편집자 및 이사, Pharmaceutical Statistics의 부편집장 및 편집장, 그리고 현재 Pharmaceutical Statistics 및 Journal of Pharmacokinetics and Pharmacodynamics의 자문 위원 활동이 포함됩니다.

Andreas는 1987년에 S 언어를 사용하기 시작했고, 1988년에 유럽 S-PLUS 배포자 및 지원자가 되었으며, 1990년대에 R을 사용하기 시작했습니다. 그는 The Basics of S and S-Plus (Springer, 4판, 20,000부 이상 판매)의 저자이며 A picture is worth a thousand tables: graphics in life sciences (Springer, 2012)의 편집자입니다.

그래픽과 시각화는 그에게 오랜 관심 주제였으며, 다양한 컨퍼런스와 세미나에서 "임상 데이터의 그래픽: 좋은 것, 나쁜 것, 그리고 추한 것"에 대한 수많은 발표와 워크숍을 진행했습니다.

- LinkedIn: linkedin.com/in/andreaskrause
- ORCID: orcid.org/0000-0002-4686-976X

9.2 Nicola Rennie

Nicola Rennie는 데이터를 효과적으로 전달하는 데 열정을 가진 학자, 데이터 과학자, 교육자입니다. 그녀는 랭커스터 의과대학(Lancaster Medical School)의 보건 데이터 과학 강사로 재직 중이며, 건강 결과를 개선하기 위한 통계 모델 사용을 연구하고 있습니다. 그녀의 강의 분야는 데이터 시각화, R 및 Python 프로그래밍, 통계 분석 결과를 효과적으로 전달하는 방법 등을 다룹니다. Nicola는 여러 R 패키지의 저자이자 관리자이며, R 사용자 그룹 모임에서 정기적으로 발표하고 커뮤니티의 새로운 구성원을 멘토링하는 등 R 커뮤니티의 활동적인 구성원입니다.

- 웹사이트: nrennie.rbind.io
- BlueSky: nrennie.bsky.social
- LinkedIn: linkedin.com/in/nicola-rennie
- Mastodon: fosstodon.org/@nrennie
- ORCID: orcid.org/0000-0003-4797-557X

9.3 Brian Tarran

Brian Tarran은 연구 및 데이터 분야를 20년 동안 다뤄온 작가이자 편집자입니다. 그는 2014년부터 2024년까지 영국 왕립 통계 학회(RSS)에서 근무했으며, 2022년 10월 Real World Data Science를 런칭하기 전까지 Significance 매거진(RSS, 미국 통계 협회, 호주 통계 학회 공동 간행물)의 편집자였습니다. Brian은 Research-Live.com의 전 편집자였으며, 시장 조사 협회(Market Research Society)에서 발행하는 Impact 매거진의 창간 편집자였습니다.

- Twitter: @brtarran
- LinkedIn: linkedin.com/in/brian-tarran-58b0261
- Mastodon: @mastodon.social/@brtarran
- GitHub: github.com/brtarran

9.4 기여자

- Jose Storopoli가 RSS 간행물을 위한 스타일링에 Julia 예제를 추가했습니다.

9.5 감사의 말

이 가이드의 초기 버전에 대한 코멘트를 주신 Kelly Zou, Marco Geraci, Jordi Prats-Rodriguez, Marc Vandemeulebroecke, Peter Bonate에게 감사드립니다. 또한 다음 분들께 감사드립니다:

- 스타일링 수정을 해주신 Philip Shirk.
- 버그 탐색을 해주신 Brent Riechelman.
- 오류 발견을 해주신 MMJansen.
- 참고문헌에 DOI 추가 및 누락된 라벨 발견을 해주신 Christine P. Chai.

10 이용 약관

10.1 법적 면책 조항

이 웹사이트에 게시된 사실 및 의견 진술은 각각의 저자 및 기여자의 것이며 영국 왕립 통계학회(RSS)의 입장과는 다를 수 있습니다.

저자들은 이 웹사이트의 콘텐츠를 책임감 있고 신중하게 준비했습니다. 그러나 저자와 RSS는 이 웹사이트나 다른 링크된 웹사이트 또는 그 이후의 링크에 포함된 정보의 정확성에 대해 명시적이든 묵시적이든 어떠한 보증도 부인합니다. 여기에는 다음이 포함되지만 이에 국한되지는 않습니다:

- 상품성 및 특정 목적에의 적합성에 대한 묵시적 보증.
- 정보를 사용하거나 정보 부족으로 인해 발생하는 컴퓨터 하드웨어, 데이터, 정보, 자료 및 비즈니스에 대한 손해에 대한 책임.
- 정보의 오류, 누락 또는 부정확성.
- 정보에 의존하여 내린 결정이나 취한 행동 또는 취하지 않은 행동.

저자와 RSS는 정보의 내용, 정확성, 시의성 또는 완전성에 대해 어떠한 보증도 하지 않으며, 어떠한 이유로든 정보에 의존할 수 있다는 보증을 하지 않으며, 링크된 사이트나 그 이후의 링크에 대한 정확성, 내용 또는 합법성에 대해 어떠한 책임도 지지 않습니다. 저자와 RSS는 웹사이트 서비스가 중단되지 않거나 오류가 없으며 결함이 수정될 수 있다는 어떠한 보증도 하지 않습니다.

저자와 RSS는 이 웹사이트의 정보 사용 또는 의존, 또는 인터넷 사용으로 인해 발생하는 어떠한 손실이나 손해(결과적 손실 또는 손해를 포함하되 이에 국한되지 않음)에 대해서도 책임을 지지 않습니다. 다른 웹사이트에 대한 링크나 광고 게재는 저자와 RSS의 보증이나 승인을 구성하지 않습니다.

이러한 면책 조항 및 제외 사항은 잉글랜드 및 웨일스 법률의 적용을 받으며 잉글랜드 및 웨일스 법원의 전속 관할권에 따릅니다. 영국 외부에서 이 사이트에 접속하기로 선택한 사람은 현지 법률이 적용되는 경우 현지 법률을 준수할 책임이 있습니다.

이 사이트를 사용함으로써 귀하는 이러한 이용 약관에 동의하게 됩니다.

10.2 사이트 콘텐츠

이 가이드의 저작권은 © 2023 Andreas Krause, Nicola Rennie 및 영국 왕립 통계 학회(Royal Statistical Society)에 있습니다. 달리 명시되지 않는 한, 이 가이드의 모든 저작권 콘텐츠는 Creative Commons Attribution 4.0 (CC BY 4.0) International 라이선스 하에 배포됩니다. 이는 원저작자를 표시하는 한, 어떤 목적으로든 사용 및 수정이 가능함을 의미합니다.

우리는 저작권 소유자를 찾고, 연락하고, 인정하기 위해 모든 합리적인 노력을 기울이고 있으며, 이 웹사이트에서 적절하게 식별되고 인정되지 않은 저작권 소유자가 있다면 필요한 수정 조치를 취할 수 있도록 알려주시기를 바랍니다.

10.3 우리가 링크하는 웹사이트는?

저자와 기여자는 사용자에게 적합하고 유용하다는 판단에 따라 외부 웹 링크를 추천합니다. 링크 선택 및 웹사이트 추가는 전적으로 저자의 문제이며 저자만의 권한입니다.

상호 링크에 대한 계약을 체결하는 것은 우리의 정책이 아닙니다.

조직이나 개인의 웹사이트에 대한 링크를 포함한다고 해서 해당 조직이나 개인의 제품, 서비스, 정책 또는 의견에 대한 저자와 RSS의 보증이나 승인을 구성하는 것은 아닙니다. 저자와 RSS는 외부 웹사이트의 콘텐츠에 대해 책임을 지지 않습니다.

10.4 우리가 링크하지 않을 웹사이트는?

우리는 인종 차별적, 성적 또는 오해의 소지가 있는 콘텐츠를 포함하거나, 폭력을 조장하거나, 영국 법률을 위반하거나, 개인이나 집단에게 불쾌감을 주는 웹사이트에는 링크하지 않습니다.

저자의 결정은 최종적이며 이에 대한 서신 교환은 하지 않습니다.

우려 사항을 제기하려면 a.britten@rss.org.uk로 이메일을 보내주십시오.

10.5 소프트웨어 및 서비스

이 사이트의 소스 코드와 파일은 GitHub에서 확인할 수 있습니다. GitHub 리포지토리의 사용은 기여자 서약 행동 강령의 적용을 받습니다.

이 사이트는 Posit에서 개발한 오픈 소스 과학 및 기술 출판 시스템인 Quarto를 사용하여 구축되었습니다. Quarto 소스 코드 및 소프트웨어 라이선스는 GitHub에서 확인할 수 있습니다.

이 사이트는 GitHub Pages에서 호스팅됩니다.

이 사이트는 웹 분석 보고를 위해 Google Analytics 4를 사용합니다.

10.6 게시 중단 정책

권리 보유자로서 허가하지 않았거나 국내법의 제한이나 예외에 해당하지 않는 자료를 사이트에서 발견하여 우려되는 경우, 다음 사항을 명시하여 서면으로 문의해 주십시오:

1. 연락처 세부 정보.
2. 자료의 전체 서지 정보.
3. 자료를 발견한 정확하고 전체적인 URL.
4. 귀하가 권리 보유자라는 증명과, 위증 시 처벌을 감수하고 귀하가 권리 보유자이거나 승인된 대리인이라는 진술.

연락처 세부 정보:
Notice and Takedown,
Licensing,
12 Errol Street,
London EC1Y 8LX
web@rss.org.uk

알림을 받으면 다음과 같이 '게시 중단' 절차가 호출됩니다:

1. 이메일이나 편지로 불만 사항 접수를 확인하고 불만 사항의 타당성과 개연성에 대한 초기 평가를 수행합니다.
2. 타당한 불만 사항이 접수되면 합의된 해결책이 나올 때까지 해당 자료는 웹사이트에서 일시적으로 제거됩니다.
3. 해당되는 경우 자료를 기탁한 기여자에게 연락합니다. 기여자는 자료가 불만 사항의 대상이라는 것과 어떤 혐의인지 통보받고, 관련 불만 사항을 진정시키도록 권장받습니다.
4. 불만 제기자와 기여자는 문제를 신속하고 원만하게 해결하고 양 당사자가 만족할 수 있도록 권장받으며, 다음과 같은 결과를 도출할 수 있습니다:
    - 자료가 변경 없이 웹사이트에 다시 게시됩니다.
    - 자료가 변경되어 웹사이트에 다시 게시됩니다.
    - 자료가 웹사이트에서 영구적으로 제거됩니다.

기여자와 불만 제기자가 해결책에 동의하지 못하는 경우, 해결책에 도달할 때까지 자료는 웹사이트를 통해 사용할 수 없습니다.

10.7 기여자 서약 행동 강령

우리의 서약

회원, 기여자 및 리더로서 우리는 나이, 신체 크기, 눈에 띄거나 보이지 않는 장애, 민족, 성 특징, 성 정체성 및 표현, 경력 수준, 교육, 사회 경제적 지위, 국적, 외모, 인종, 카스트, 피부색, 종교 또는 성적 정체성 및 지향과 관계없이 커뮤니티 참여를 모두에게 괴롭힘 없는 경험으로 만들 것을 약속합니다.

우리는 개방적이고, 환영하며, 다양하고, 포용적이며, 건전한 커뮤니티에 기여하는 방식으로 행동하고 상호 작용할 것을 약속합니다.

우리의 표준

우리 커뮤니티의 긍정적인 환경에 기여하는 행동의 예는 다음과 같습니다:

- 다른 사람들에게 공감과 친절을 베푸는 것
- 서로 다른 의견, 관점 및 경험을 존중하는 것
- 건설적인 피드백을 주고 정중하게 받아들이는 것
- 우리의 실수로 영향을 받은 사람들에게 책임을 지고 사과하며, 경험을 통해 배우는 것
- 개인으로서의 우리가 아니라 커뮤니티 전체에 가장 좋은 것에 초점을 맞추는 것

용납될 수 없는 행동의 예는 다음과 같습니다:

- 성적인 언어 또는 이미지의 사용, 어떤 종류의 성적 관심이나 접근
- 트롤링, 모욕적이거나 경멸적인 댓글, 개인적 또는 정치적 공격
- 공적 또는 사적인 괴롭힘
- 물리적 주소나 이메일 주소와 같은 타인의 개인 정보를 명시적인 허가 없이 게시하는 것
- 전문적인 환경에서 부적절하다고 합리적으로 간주될 수 있는 기타 행위

집행 책임

커뮤니티 리더는 허용되는 행동의 표준을 명확히 하고 집행할 책임이 있으며, 부적절하거나 위협적이거나 불쾌하거나 해롭다고 판단되는 행동에 대해 적절하고 공정한 시정 조치를 취할 것입니다.

커뮤니티 리더는 이 행동 강령에 부합하지 않는 댓글, 커밋, 코드, 위키 편집, 이슈 및 기타 기여를 제거, 편집 또는 거부할 권리와 책임이 있으며, 적절한 경우 중재 결정의 이유를 전달할 것입니다.

범위

이 행동 강령은 모든 커뮤니티 공간(이 사이트 및 GitHub 리포지토리 포함) 내에서 적용됩니다. 또한 개인이 공적 공간에서 커뮤니티를 공식적으로 대표할 때도 적용됩니다. 커뮤니티를 대표하는 예로는 공식 이메일 주소 사용, 공식 소셜 미디어 계정을 통한 게시, 또는 온/오프라인 이벤트에서 임명된 대표자로서 활동하는 것 등이 있습니다.

RSS와의 서면 합의에 의한 사전 허가가 없는 한, 편집팀(Andreas Krause, Nicola Rennie, Anna Britten)만이 커뮤니티를 공식적으로 대표할 수 있음을 유의하십시오. 언론에 대한 논평은 임명된 대표자만 할 수 있으며 RSS 공보실의 승인을 받아야 합니다.

집행

학대, 괴롭힘 또는 기타 용납될 수 없는 행동의 사례는 집행 책임이 있는 커뮤니티 리더에게 a.britten@rss.org.uk로 신고할 수 있습니다. 모든 불만 사항은 신속하고 공정하게 검토 및 조사될 것입니다.

모든 커뮤니티 리더는 사건 신고자의 개인 정보와 보안을 존중할 의무가 있습니다.

집행 지침

커뮤니티 리더는 이 행동 강령을 위반했다고 판단되는 조치에 대한 결과를 결정할 때 다음 커뮤니티 영향 지침을 따를 것입니다:

1. 교정

커뮤니티 영향: 커뮤니티에서 부적절하거나 환영받지 못하는 언어 또는 기타 행동의 사용.

결과: 커뮤니티 리더의 비공개 서면 경고. 위반의 성격을 명확히 하고 왜 그 행동이 부적절했는지 설명합니다. 공개 사과가 요청될 수 있습니다.

2. 경고

커뮤니티 영향: 단일 사건 또는 일련의 조치를 통한 위반.

결과: 지속적인 행동에 대한 결과를 포함한 경고. 관련된 사람들과의 상호 작용 금지(행동 강령을 집행하는 사람들과의 원치 않는 상호 작용 포함)가 일정 기간 동안 적용됩니다. 여기에는 커뮤니티 공간뿐만 아니라 소셜 미디어와 같은 외부 채널에서의 상호 작용 회피도 포함됩니다. 이 조건을 위반하면 일시적 또는 영구적 차단으로 이어질 수 있습니다.

3. 일시적 차단

커뮤니티 영향: 지속적인 부적절한 행동을 포함한 커뮤니티 표준의 심각한 위반.

결과: 일정 기간 동안 커뮤니티와의 모든 종류의 상호 작용 또는 공개 통신의 일시적 차단. 이 기간 동안 관련된 사람들과의 공개적 또는 사적인 상호 작용(행동 강령을 집행하는 사람들과의 원치 않는 상호 작용 포함)은 허용되지 않습니다. 이 조건을 위반하면 영구적 차단으로 이어질 수 있습니다.

4. 영구적 차단

커뮤니티 영향: 지속적인 부적절한 행동, 개인에 대한 괴롭힘, 또는 특정 집단에 대한 공격이나 비하를 포함한 커뮤니티 표준 위반 패턴의 입증.

결과: 커뮤니티 내 모든 종류의 공개 상호 작용 영구 차단.

출처

이 행동 강령은 Contributor Covenant, 버전 2.1에서 수정되었습니다. https://www.contributor-covenant.org/version/2/1/code_of_conduct.html에서 확인할 수 있습니다.

커뮤니티 영향 지침은 Mozilla의 행동 강령 집행 사다리에서 영감을 받았습니다.

이 행동 강령에 대한 일반적인 질문에 대한 답변은 https://www.contributor-covenant.org/faq의 FAQ를 참조하십시오. 번역본은 https://www.contributor-covenant.org/translations에서 확인할 수 있습니다.