이제 이 페이지를 편집할 수 있습니다. 편집 완료

Ch3&4. Describing Data & Continuous Vars

발행일	2022년 1월 15일
상태	Post
태그	study

댓글 추가

Check-in (10min)

- 지난 한 주는 어땠나요? 다섯글자로 표현하고 그 이유를 설명해주세요.
    - 중장기계획, 너무바빴음, 롤러코스터 or 지옥과천국, 백신맞은주, 샤이니만세 (ft. RShiny), 휴가후여파 & 티끌모아산, 이럴줄알았 → 새해엔다들

Chapter3. Describing Data (15min)

3.1 Simulating data

3.1.1 Store Data: Setting the Structure

- data frame
- var type: int, float, logi, chr, factor

- 주요변수
  - 사이즈: dim()
  - 미리보기: str(), head(), tail(), some()
  - 변환: factor() chr → factor
    - categorical var. → dummy coding

3.1.2 Store Data: Simulating Data Points

- set.seed(###)
- rbinom(n, size, p): random binomial
- rpois(n, lambda): Poisson distribution

3.2 Functions to Summarize a Variable

3.2.1 Discrete Variables

- table

3.2.2 Continuous Variables

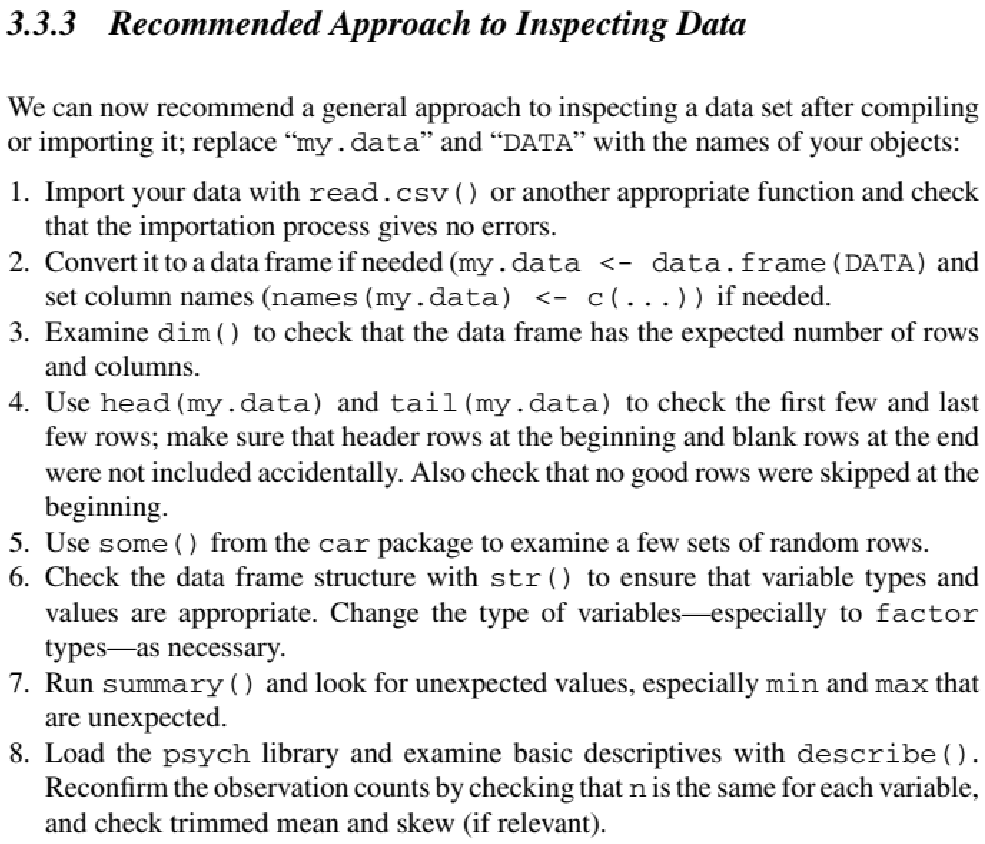
- min, max, ..., quantile

3.3 Summarizing Data Frames

3.3.1 summary()

3.3.2 describe()

3.3.3 Recommended Approach to Inspecting Data



3.3.3 Recommended Approach to Inspecting Data

We can now recommend a general approach to inspecting a data set after compiling or importing it; replace “`my.data`” and “`DATA`” with the names of your objects:

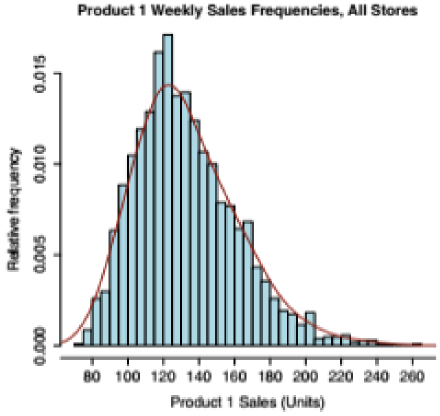
1. Import your data with `read.csv()` or another appropriate function and check that the importation process gives no errors.
2. Convert it to a data frame if needed (`my.data <- data.frame(DATA)` and set column names (`names(my.data) <- c(...)`) if needed.
3. Examine `dim()` to check that the data frame has the expected number of rows and columns.
4. Use `head(my.data)` and `tail(my.data)` to check the first few and last few rows; make sure that header rows at the beginning and blank rows at the end were not included accidentally. Also check that no good rows were skipped at the beginning.
5. Use `some()` from the `car` package to examine a few sets of random rows.
6. Check the data frame structure with `str()` to ensure that variable types and values are appropriate. Change the type of variables—especially to `factor` types—as necessary.
7. Run `summary()` and look for unexpected values, especially `min` and `max` that are unexpected.
8. Load the `psych` library and examine basic descriptives with `describe()`. Reconfirm the observation counts by checking that `n` is the same for each variable, and check trimmed mean and skew (if relevant).

3.3.4 apply()

3.4 Single Variable Visualization

3.4.1 Histograms

Fig. 3.6 Final histogram with density curve



3.4.2 Boxplots

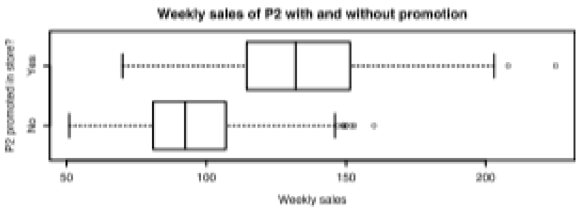


Fig. 3.9 Boxplot of product sales by promotion status

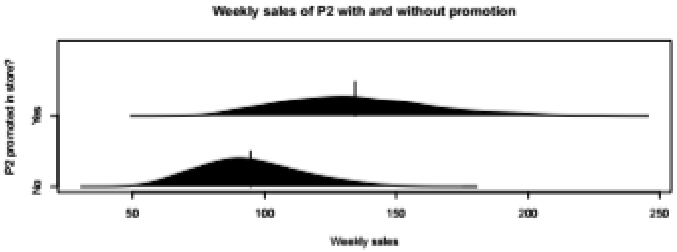


Fig. 3.10 Beanplot of product sales by promotion status

3.4.3 QQ Plot to Check Normality

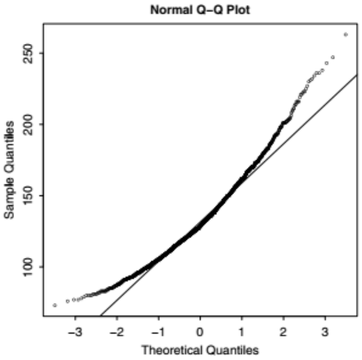


Fig. 3.11 QQ plot to check distribution. The tails of the distribution bow away from the line that represents an exact normal distribution, showing that the distribution of `p1sales` is skewed

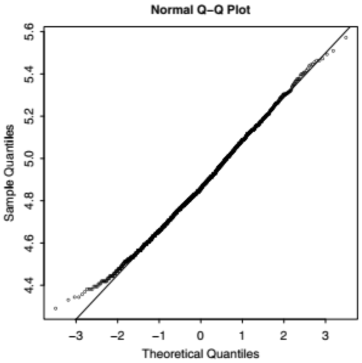


Fig. 3.12 QQ plot for the data after `log()` transformation. The sales figures are now much better aligned with the solid line that represents an exact normal distribution

```
> qqnorm(log(store.df$p1sales))
> qqline(log(store.df$p1sales))
```

3.4.4 Cumulative Distribution

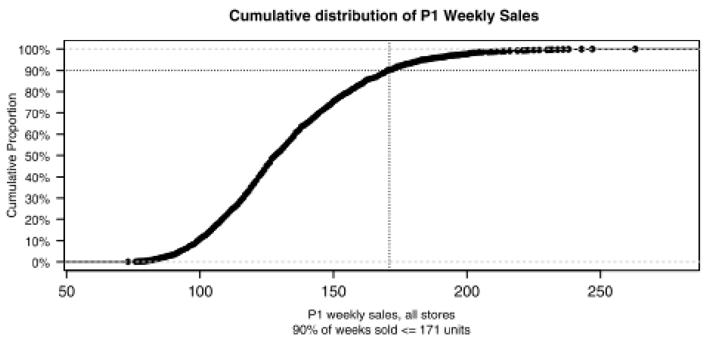


Fig. 3.13 Cumulative distribution plot with lines to emphasize the 90th percentile. The chart identifies that 90% of weekly sales are lower than or equal to 171 units. Other values are easy to read off the chart. For instance, roughly 10% of weeks sell less than 100 units, and fewer than than 5% sell more than 200 units

3.4.5 Language Brief: by() and aggregate()

- 그룹별로 연산 → melt / reshape 을 하는 것보다 훨씬 쉽다!
    - by(data=DATA, indices=INDICES, fun=FUNCTION)
    - aggregate(data=DATA, by=BY, fun=FUNCTION)

3.4.6 Maps

- choropleth

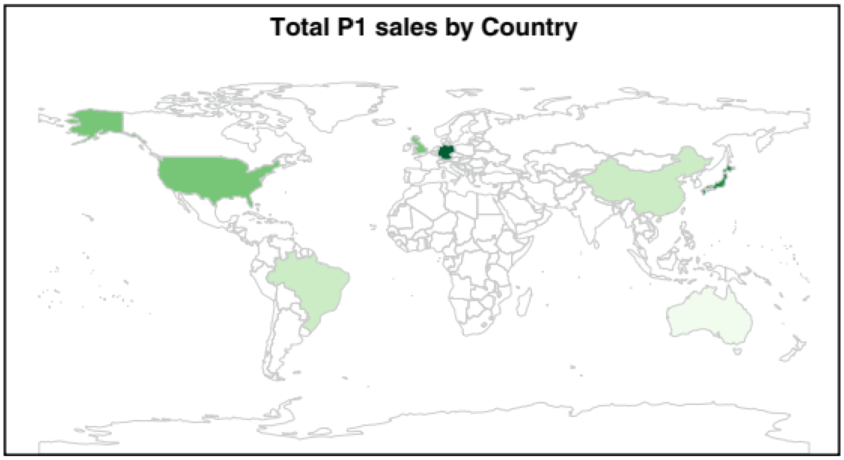
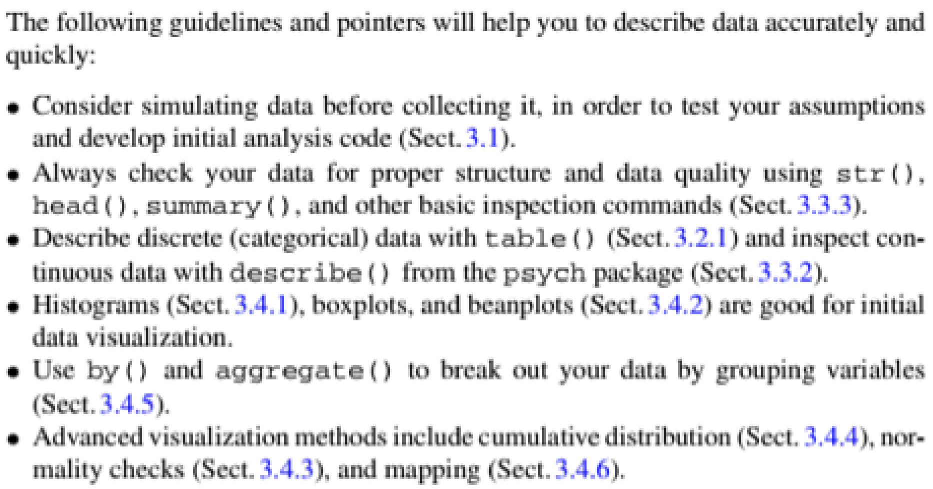


Fig. 3.14 World map for P1 sales by country, using `rworldmap`

3.5 KeyPoints



The following guidelines and pointers will help you to describe data accurately and quickly:

- Consider simulating data before collecting it, in order to test your assumptions and develop initial analysis code (Sect. 3.1).
- Always check your data for proper structure and data quality using `str()`, `head()`, `summary()`, and other basic inspection commands (Sect. 3.3.3).
- Describe discrete (categorical) data with `table()` (Sect. 3.2.1) and inspect continuous data with `describe()` from the `psych` package (Sect. 3.3.2).
- Histograms (Sect. 3.4.1), boxplots, and beanplots (Sect. 3.4.2) are good for initial data visualization.
- Use `by()` and `aggregate()` to break out your data by grouping variables (Sect. 3.4.5).
- Advanced visualization methods include cumulative distribution (Sect. 3.4.4), normality checks (Sect. 3.4.3), and mapping (Sect. 3.4.6).

3.6 Data Sources

- SQL → CSV or spreadsheet → CSV

3.7. Learning More

- Plotting: lattice, ggplot2
- Maps: rworldmap, ggplot2, ggmap

Ch3. Discussion

- 데이터 과학자는 작업시간의 80% 를 전처리에 소비한다고 합니다. 그만큼 raw data 를 다루는 게 손이 많이 가는 작업입니다. 다루어보신 데이터 중에 가장 까다롭거나 복잡한 데이터는 어떤 것이었나요? 해당 데이터를 잘 "Describe" 하는 데에 효과적인 방법이 있었나요?

Chapter4. Relationships Between Continuous Variables (15min)

4.1 Retailer Data

4.1.1 Simulating the Data

4.1.2 Simulating Online and In-store Sales Data

4.1.3 Simulating Satisfaction Survey Responses

4.1.4 Simulating Non-response Data

4.2 Exploring Associations Between Variables with Scatterplots

4.2.1 Creating a Basic Scatterplot with plot()

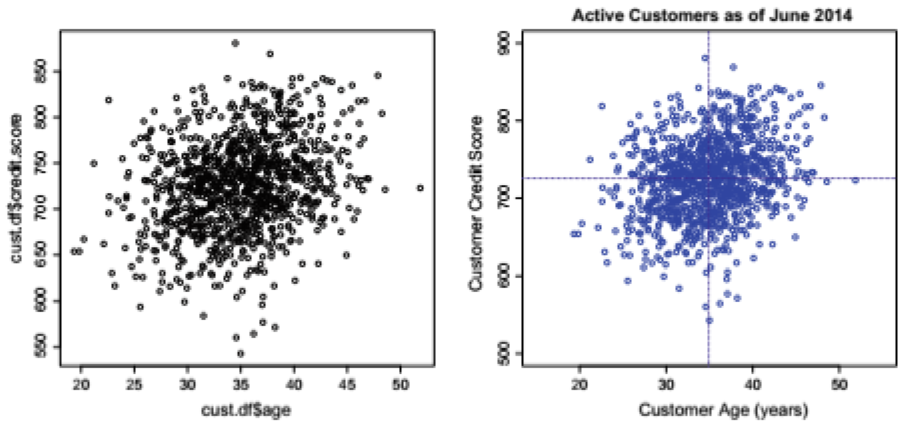
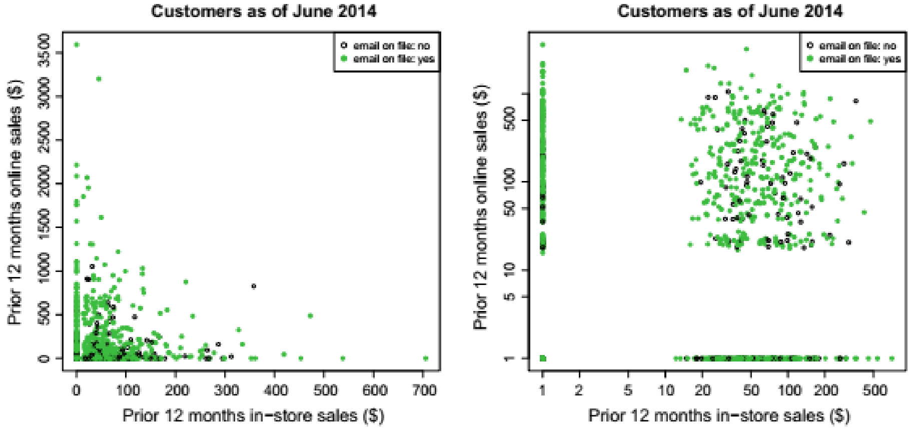


Fig. 4.1 Basic scatterplot of customer age versus credit score using default settings in `plot()` function (left), and a properly labeled version of the same plot (right)

4.2.2 Color-Coding Points on a Scatterplot

4.2.3 Adding a Legend to a Plot



4.2.4 Plotting on a Log Scale

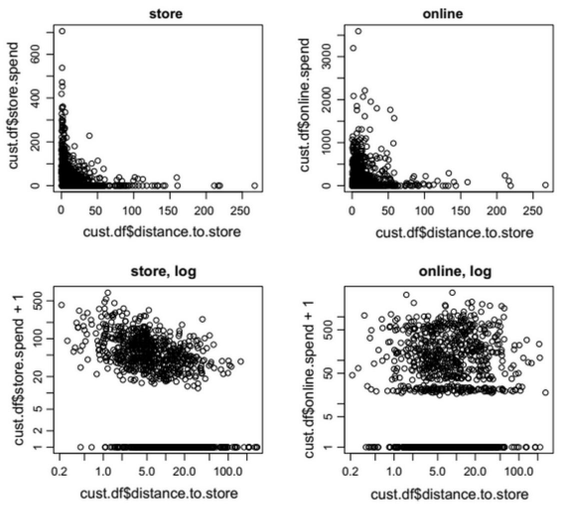


Fig. 4.5 A single graphic object consisting of multiple plots shows that distance to store is related to in-store spending, but seems to be unrelated to online spending. The relationships are easier to see when spending and distance are plotted on a log scale using `log="xy"` in the two lower panels

4.3 Combining Plots in a Single Graphics Object

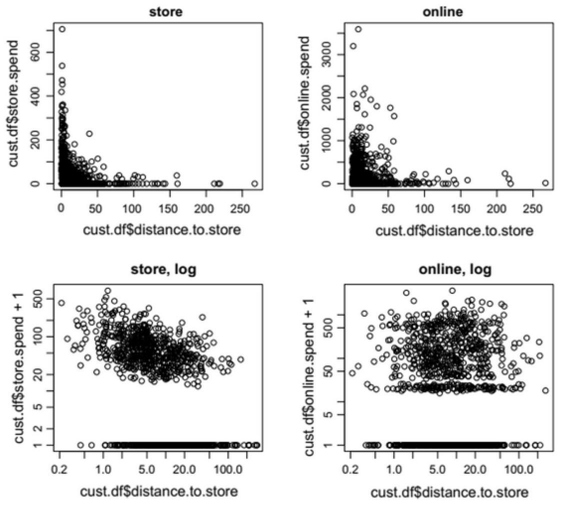


Fig. 4.5 A single graphic object consisting of multiple plots shows that distance to store is related to in-store spending, but seems to be unrelated to online spending. The relationships are easier to see when spending and distance are plotted on a log scale using `log="xy"` in the two lower panels

4.4 Scatterplot Matrices

4.4.1 pairs()

4.4.2 scatterplotMatrix()

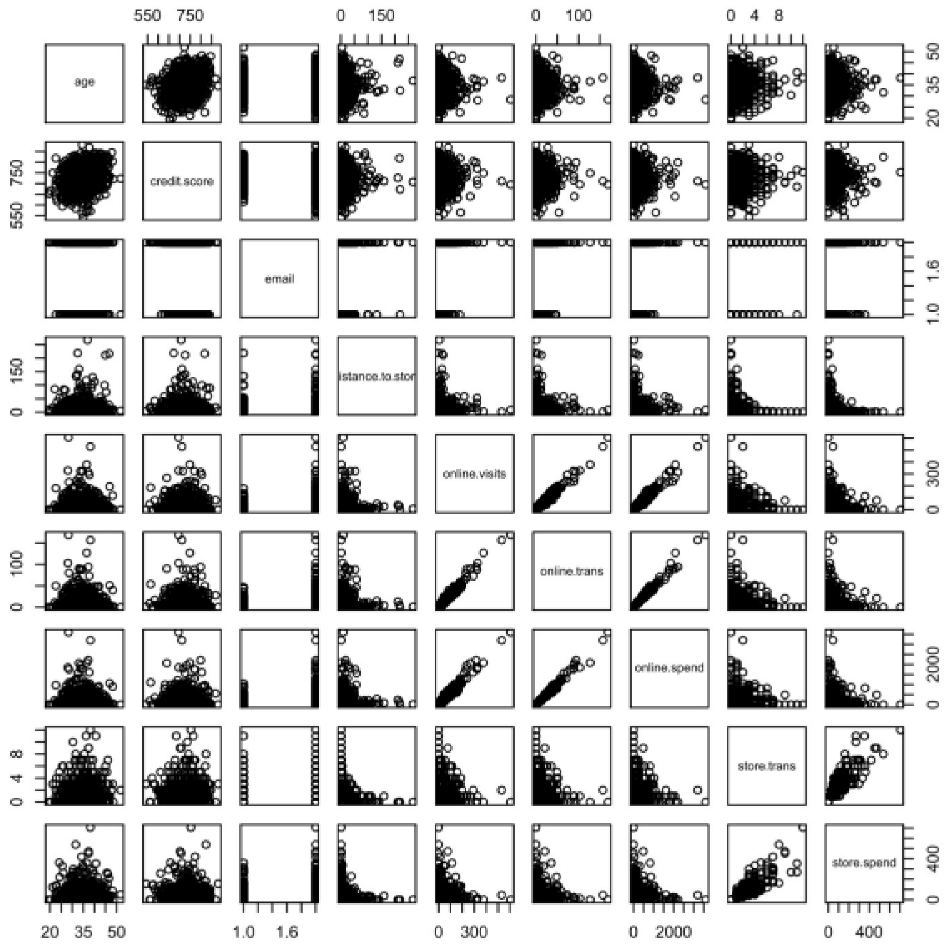


Fig. 4.6 A scatterplot matrix for the customer data set produced using `pairs()`

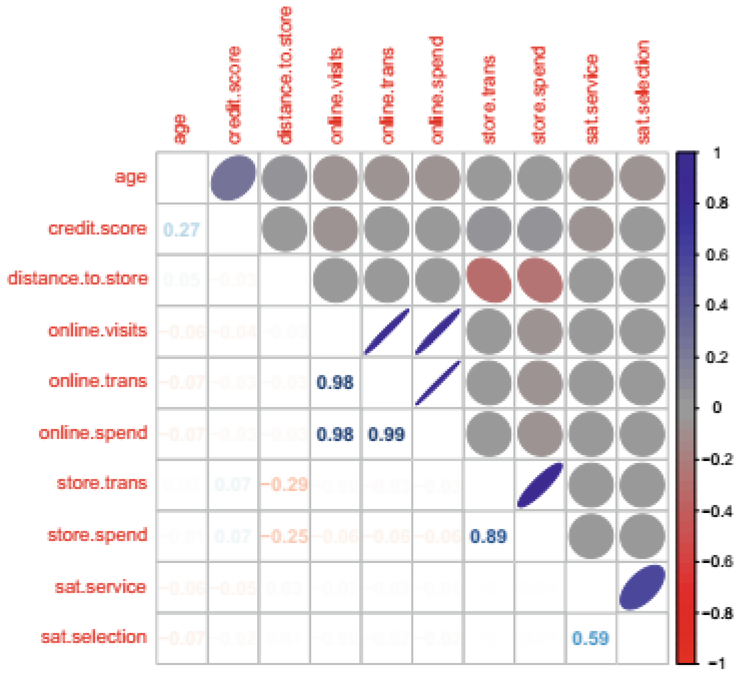
4.5 Correlation Coefficients

4.5.1 Correlation Tests

- cor(x, y)

4.5.2 Correlation Matrices

Fig. 4.8 A correlation plot produced using `corrplot.mixed()` from the `corrplot` package is an easy way to visualize all of the correlations in the data. Correlations close to zero are plotted as circular and gray (using the color scheme we specified), while magnitudes away from zero produce ellipses that are increasingly tighter and blue for positive correlation and red for negative



4.5.3 Transforming Variables Before Computing Correlations

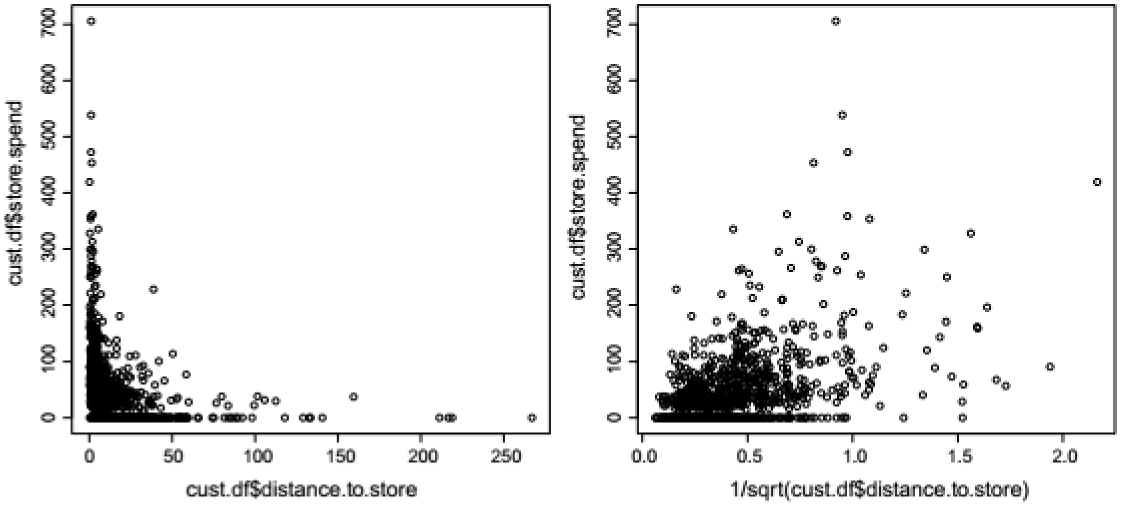


Fig. 4.9 A transformation of `distance.to.store` to its inverse square root makes the association with `store.trans` more apparent in the right-hand chart, as compared to the original values on the left

4.5.4 Typical Marketing Data Transformations

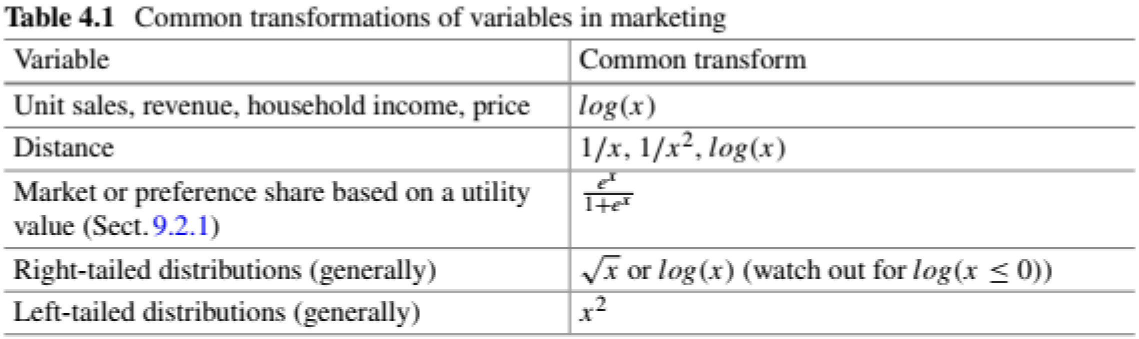


Table 4.1 Common transformations of variables in marketing

Variable	Common transform
Unit sales, revenue, household income, price	$log(x)$
Distance	$1/x$, $1/x^2$, $log(x)$
Market or preference share based on a utility value (Sect. 9.2.1)	$\frac{e^x}{1+e^x}$
Right-tailed distributions (generally)	$\sqrt{x}$ or $log(x)$ (watch out for $log(x \leq 0)$)
Left-tailed distributions (generally)	x^2

4.5.5 Box-Cox Transformations

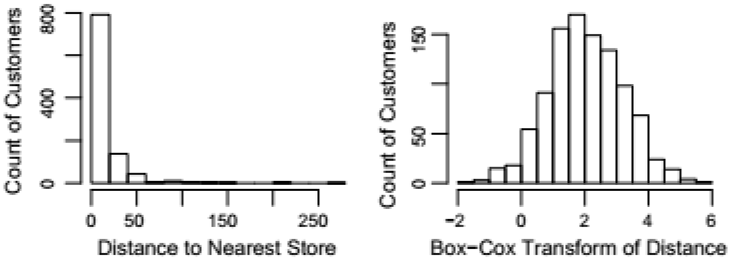


Fig. 4.10 A Box-Cox transformation of `distance.to.store` makes the distribution closer to Normal

4.6 Exploring Associations in Survey Responses

4.6.1 jitter()

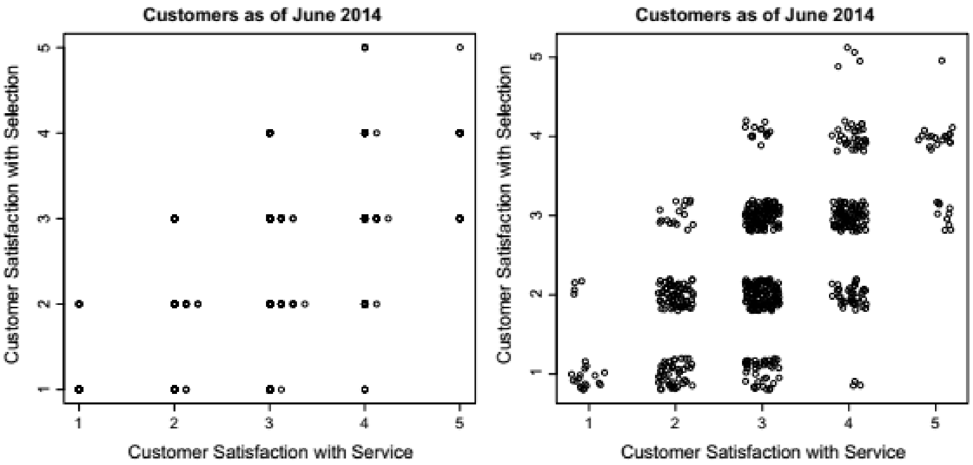


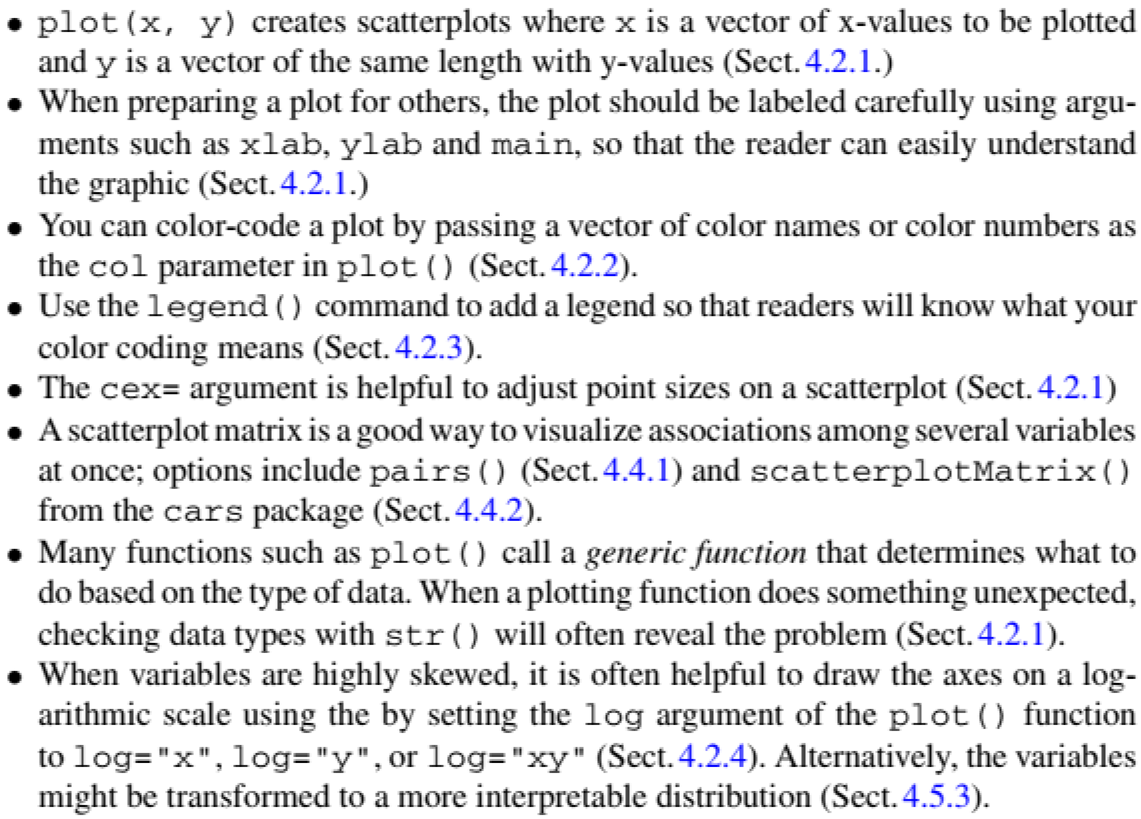
Fig. 4.11 A scatter plot of responses on a survey scale (left) is not very informative. Using jitter (right) makes the plot more informative and reveals the number of observations for each pair of response values

4.6.2 polychoric()

- Ordinal var (c.f. Likert)

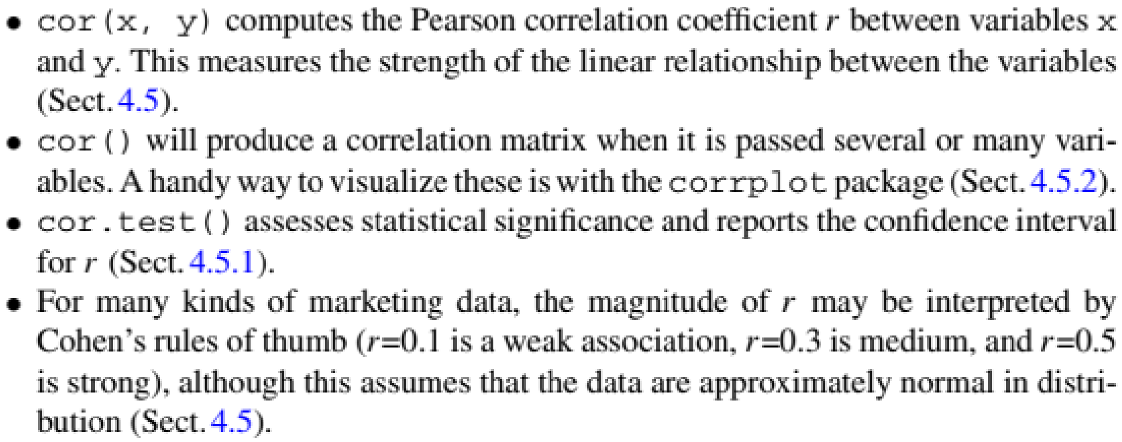
4.7 Key Points

- Visualization



- `plot(x, y)` creates scatterplots where `x` is a vector of x-values to be plotted and `y` is a vector of the same length with y-values (Sect. 4.2.1.)
- When preparing a plot for others, the plot should be labeled carefully using arguments such as `xlab`, `ylab` and `main`, so that the reader can easily understand the graphic (Sect. 4.2.1.)
- You can color-code a plot by passing a vector of color names or color numbers as the `col` parameter in `plot()` (Sect. 4.2.2).
- Use the `legend()` command to add a legend so that readers will know what your color coding means (Sect. 4.2.3).
- The `cex=` argument is helpful to adjust point sizes on a scatterplot (Sect. 4.2.1)
- A scatterplot matrix is a good way to visualize associations among several variables at once; options include `pairs()` (Sect. 4.4.1) and `scatterplotMatrix()` from the `cars` package (Sect. 4.4.2).
- Many functions such as `plot()` call a *generic function* that determines what to do based on the type of data. When a plotting function does something unexpected, checking data types with `str()` will often reveal the problem (Sect. 4.2.1).
- When variables are highly skewed, it is often helpful to draw the axes on a logarithmic scale using the by setting the `log` argument of the `plot()` function to `log="x"`, `log="y"`, or `log="xy"` (Sect. 4.2.4). Alternatively, the variables might be transformed to a more interpretable distribution (Sect. 4.5.3).

- Statistics



- `cor(x, y)` computes the Pearson correlation coefficient *r* between variables `x` and `y`. This measures the strength of the linear relationship between the variables (Sect. 4.5).
- `cor()` will produce a correlation matrix when it is passed several or many variables. A handy way to visualize these is with the `corrplot` package (Sect. 4.5.2).
- `cor.test()` assesses statistical significance and reports the confidence interval for *r* (Sect. 4.5.1).
- For many kinds of marketing data, the magnitude of *r* may be interpreted by Cohen's rules of thumb (r=0.1 is a weak association, r=0.3 is medium, and r=0.5 is strong), although this assumes that the data are approximately normal in distribution (Sect. 4.5).

4.8 Data Sources

- merge()

4.9 Learning More

- Plotting: ggplot2, lattice

- Correlation analysis
- Analyzing survey scale responses: polychor(), bayesm, rscaleUsage()

Ch4. Discussion

- Visualization을 통해 여러 변수 간의 새로운 insight를 얻거나 해결의 실마리를 찾은 경험이 있으신가요?
    - 충성고객은 오히려 사이트에 체류하는 시간이 짧았음 → “머무르게하자” 라는 편견과 반대
        - VIP → 거래액이 높은 사람들. 빠르게 혜택을 주는 게 더 좋음.

QnA

- 실무에서의 데이터 분석 프로세스?
    - 정형화되어 있지 않음.
    - 마케터: “매출이 더 나오려면 어떻게 해야해요?” → 통계적인 언어로 변환해서 regressions 등 수단들을 찾음
- 경영적인 판단을 위한 데이터 분석
    - 같은 데이터도 “목적” 에 따라서 다르게 해석 → 분석 방법보다 “질문”이 중요
        - ex) 비용을 줄이고 싶은지, 주력 상품을 선택하고 싶은지 등
    - 정형화된 데이터 수집이 이루어지고 있어서 새로운 데이터를 수집하고 분석법을 실행하는데에 어려움
    - “시각화” → 커뮤니케이션 도구. 내부 (보고, 유관부서 소통), 외부(고객)에도 정말 중요 ★★★
- 제품/서비스의 속성에 따라서 필요한 데이터의 종류와 분석이 달라짐
    - 제품 vs. 서비스
    - 장기 서비스 (보험 등) vs. 단기 서비스

- 시각화
  - Sankey diagram
  - Sunburst diagram
    - https://www.r-graph-gallery.com/circular-barplot.html

참고자료

- Box-Cox
  - https://m.blog.naver.com/jiehyunkim/220616091027
- 시각화 예시
  - https://www.r-graph-gallery.com/
  - https://www.python-graph-gallery.com/
- 시각화 현업 적용 사례
  - 게임 개발 과정에서의 도식화 https://gamebiz.jp/news/131227

회고 (15min)

- 손경희
  - Plus 좋았던 점
    - 지난주에 비해 실제 분석 대한 이야기가 많이 나와서 책 외에 새로 배운 것이 많았습니다
    - 제가 스터디하면서 이해가 어려웠던 부분을 알려주셔서 좋았습니다
  - Minus 아쉬웠던 점
    - 현업에서의 사례를 더 들어보고 싶었어요
  - Insight 배운 점
    - 서로 다른 관점이 존중될 때 시너지가 나네요

- 박규서
  - Plus 좋았던 점
    - 토요일 아침 좋은 분들과 새로운 만남을 가진 점
    - 여러 좋은 말씀과 새로운 item에 대하여 듣게 된 점
  - Minus 아쉬웠던 점
    - 시간이 짧은 점
  - Insight 배운 점
    - 역시 집단 지성이 좋다는 점
    - 내가 모르는 분야가 많다는 재인식
- 채충일
  - Plus 좋았던 점
    - 날이 좋아서, 날이 좋지 않아서, 날이 적당해서 모든날이 좋았다… (도깨비)
  - Minus 아쉬웠던 점
    - 시간이… 짧다…
  - Insight 배운 점
    - 현장에서의 케이스들과 실무자의 해석/관점, 다른관점의 환기
- 윤승원
  - Plus 좋았던 점
    - Hearing applications in practice
  - Minus 아쉬웠던 점
    - None
  - Insight 배운 점
    - What to consider, making business cases before analysis

- 강동오
    - Plus 좋았던 점
        - R로 구현 가능한 영역과 개념에 대한 이해, 다른 사례와 의견을 통한 사고의 확장
        - 손 대표님이 창업가셔서 그런지 와꾸(Structure) 및 Leading이 좋아서 계속 유사한 방향으로 진행해도 좋을 듯 합니다.
    - Minus 아쉬웠던 점
        - 전문지식이 부족하여 내용을 완벽하게 이해하지 못한 것 같아 아쉬웠습니다.
    - Insight 배운 점
        - Sankey Chart 등 GA/AA를 통해 수집한 데이터도 R을 통해 더 많은 확장성과 인사이트를 얻을 수 있다는 점
- 이승희
    - Plus 좋았던 점
        - 명쾌한 요약, 풍부한 현장사례가 좋았습니다.
    - Minus 아쉬웠던 점
        - 저는 실습을 많이 해 봐야할 듯 합니다.아직은 개념 익히기 수준입니다.
    - Insight 배운 점
        - 개떡같은 가설이라도 가설이 있어야 찰떡같은 반박이 나올 수 있다
- 이진재
    - Plus 좋았던 점
        - 노션에서 정리가 깔끔해서 좋았다. 진행도 매끄러웠다
    - Minus 아쉬웠던 점
        - 특별히 없었지만, 개인적으로 좀더 미리 책을 읽어서 준비해야 할것 같다
    - Insight 배운 점
        - 동일한 주제로 다양한 분야에서 활용하는 이야기를 들을수 있어서 참고가 되었다

- 정시앙
    - Plus 좋았던 점
        - 다른 분들 사례 들을 수 있어서 좋았다
    - Minus 아쉬웠던 점
        - 리스트
    - Insight 배운 점
        - 가설이 틀릴수도 있으니 가설을 잘 확인하고 검증하는 절차가 필요할 것 같다