🏆

자료분석 및 실습

Lecture 1. Introduction

1.1. What is Data Science?

Era of Big Data

-over the centuries, as data became larger, machines were introduced to speed up the tabulations

-in the late 19th century, statistical methods began to develop rapidly → 계산 편리성을 위한 approximation 찾고자 함

-nowdays: data sets are so large that can be processed only by machine

Misconception about Data Science

-data science ≠ data for science (data-driven research)

-data science is not just a tool for data

What is Data Science?

-science of extracting meaningful information from data

-'science': evidence를 쌓아 가는 과정

-data science is best applied in the context of expert knowledge about the domain from which the data originate

-distinction between data and information = why data science exists

-interdisciplinary field: CS - statistics - domain knowledge

-data science in industry: understand data > build prototypes > experiment …

Statistics and Data Science

-goal of statisticians and data scientists: extract meaningful information from data

-statisticians focused on creating methods that would maximize the strength of inference

Miscellaneous

-Google Flu Trend: data science failure

-Facebook Emotion Experiment 2012: manipulation → data ethics

1.2. Data Science Lifecycle

1. Question/Problem Formulation
2. Data Acquisition and Cleaning
3. Exploratory Data Analysis and Visualization

- understand the data
- 1이나 2로 돌아갈수 있음

4. Prediction and Inference

- understand the world
- data는 sample → population을 추정, 예측, 설명

1.3. Population, Sample, Bias

Sample

-sample: subset of the population

- often used to make inferences about the population
- two common sources of error: random/chance error and bias(e.g. non-response bias)

-target population vs. sampling frame: 1936 US Presidential Election

- Digest’s sample was not representative of the population: tended to vote Republican
- low response rate
- Gallup: sample size of only 3000 people

Common Biases

-selection bias: systematically excluding (or favoring) particular groups

-response bias: people don’t always respond truthfully

-non-response bias: people don’t always respond

Lecture 2. Data Visualization

2.1. Introduction to Data Visualization

What is Data Visualization?

-major part of data analysis

-allows us to explore data and find patterns

-classic example: Anscombe’s quartet → 동일한 lin.reg. / 다른 data

-’forces us to notice what we never expected to see’

Data Visualization

-no strict formula or procedure

-generally, should aim to fully explore the data and maximize what can be learned from it

Principles

-above all else show the data

-avoid distorting what the data have to say

-present many numbers in a small space etc.

Major Tools

-histogram, density plots, barplots

-boxplots

-scatterplots, bubbleplots

2.2. Example: 2012 Federal Election Cycle

-amount of money spent by candidates

- by adding variables: type of spending, House/President/Senate (office being sought)
- 공화당의 attack adds 더 많음 / H나 S에서는 민주당의 attack adds 증가

-amount of donations

- number of donations → density, General/Primary

- General에선 거의 동일, 경선에서는 Obama가 적은 금액의 donation이 더 많음

-more money spent, more votes?

- relationship very weak → misleading: 지역별 강세 후보 존재하는 곳에는 큰 돈을 주지 않음
- proportion of vote → spent 커짐

2.3. Composing Data Graphics: Taxonomy for Data Graphics

Visual Cue

-position > length > angle > direction > shapes > area > volume > color saturation > color hue

-good at accurately perceiving differences in position → “don’t use pie charts!”

-usually hard to perceive differences in color

-avoid word clouds too

Coordinate Systems

-how are data points organized?

-Cartesian, polar, geographic

Scale

-numeric: linear, logarithmic, percentage

-categorical: may be ordinal or not

-time: different units, wrap-around scale

Context

-titles, subtitles, axis labels, reference points or lines

2.4. Directory of Visualization

Amounts

-using bars or dots

-single distribution: histogram, density plot, cumulative density, quantile-quantile plot

-multiple distribution: boxplots, violins, strip charts, sina plots, stacked histograms, overlapping densities, ridgeline plots

- ridgeline plots can be a useful alternative to violin plots

Proportions

-pie chart, bars, stacked bars

-comparing proportions: multiple pie charts, grouped bars, stacked bars, stacked densities

-multiple proportions: mosaic plot, treemap, parallel sets

x-y Relationships

-scatterplot, bubble chart(3 quantitative variables), paired scatterplot, slopegraph

-large number of points can become uninformative(overplotting) → density contours, 2D bins, hex bins, correlogram(more than 2 quantities)

-line graph, connected scatterplot, smoothline graph

Geospatial Data

-map, choropleth, cartogram, cartogram heatmap

2.5. Visualizing Distributions

Histogram

-bin width h can be selected by some rules

-Scott's normal reference rule: $h = 3.49 \frac{\hat{\sigma}}{n^{1/3}}$

-Freedman-Diaconis rule: $h = 2 \frac{IQR(x)}{n^{1/3}}$

Density Curves

-kernel density estimates depend on the chosen kernel and bandwidth

-kernels: uniform, triangle, Gaussian, Cosine etc.

-bandwith selection: rule of thumb → $h = \frac{1.06}{n^{1/5}} S$ (S = standard deviation)

- can replace S by scaled interquartile range or scaled MAD

Examples

-ridgeline plots: when comparing more than two distributions

Example: Blue Jay Birds

-scatterplot: head length ~ body mass

-all-against-all matrix of scatter plots → correlograms (visualization of correlation coefficients)

2.6. Visualizing Trends

Time Series

-can fit a curve with $y = A\exp(mx)$

-log transformation of y makes a linear relationship: $\log(y) = \log(A) + mx$

Smoothing

-20-day-average → 100-day-average: more global

-moving average: median day 만큼 왼쪽으로 shift

LOESS

-LOESS(LOcally Estimated Scatterplot Smoothing)

-consider a neighbor of $x_0, N_k(x_0)$, that contains k observations

- fit a linear function $I(x) = \beta_0 + \beta_1(x - x_0)$ using k pairs of (X_i, Y_i) where $X_i \in N_k(x_0)$
- give a greater weight to X_i close to x_0
- $W(u), 0 \leq u \leq 1$ be a non-negative weighting function with mode at $u = 0$
- popular choice: $W(u) = ((1-u)^3)^3$

-weighted least squares are used to find the $I(x)$ that minimizes

$$\sum_{i:X_i \in N_k(x_0)} \{Y_i - I(X_i)\}^2 \cdot W(\frac{|x_0 - X_i|}{\triangle_{x_0}})$$

where $\triangle_{x_0} = max_{i:X_i \in N_k(x_0)}|X_i - x_0|$ is the maximum distance of x_0 to elements in $N_k(x_0)$

- need to specify the span that determines the smoothness → span = fraction of the total number of points = k/n
- larger values of span will produce smoother curves
- $I(x)$ can be replaced by a quadratic function

Splines

-LOESS requires the fitting of many separate regression models → slow for large datasets

-spline: piecewise polynomial function that is highly flexible yet looks smooth

- knots: endpoints of the individual spline segments
- spline with k segments → need to specify k+1 knots

-types of splines: cubic splines, B-splines, thin-plate splines, Gaussian process splines, etc.

2.7. Telling a Story and Making a Point

Example: Challenger

-engineers' recommendation was overruled due to a lack of persuasive evidence

-can make a point with extrapolation

Example: `nycflights13`

-mean arrival delay: bubbleplots < simple bar graph

Lecture 3. Data Wrangling

3.1. A Grammar of Data Wrangling: With One Table

Data Manipulation with `dplyr`

- `dplyr`: *grammar of data manipulation*
  - c.f. `ggplot2`: *grammar of graphics*

- `dplyr` was designed to:
  - provide commonly used data manipulation tools

- have fast performance for in-memory operations
- abstract the interface between the data manipulation operations and the data source

- `dplyr` operates on data frames and also on tibbles — trimmed-down version of data frame → designed for big data

- `print` = `head` + `str`
- `class` to give the inheritance path

Single Table Verb

- `filter()`: select rows (observations) in a data frame

- logical tests: `%in%`, `is.na`, `!is.na` / `xor`, `any`, `all`, `!` (not)
- only rows where the condition evaluates to `TRUE` are kept
- handling missing values(`NA`s)
  - 'contagious': any operation involving an unknown value will also be unknown
  - to check whether element of x is `NA`: use `is.na(x)`
  - filter out all observations with missing values: `<DATA_FRAME> %>% filter(!is.na(<VARIABLE>)`

- `select()`: select columns (variables) in a data frame

- list of columns separated by commas
- useful functions: `starts_with()`, `ends_with()`, `contains()`, `num_range("x", 1:3)`
- combining `filter()` and `select()`: drill down to specific pieces of information → `select(filter())`
  - use pipe(`%>%`) operator instead → make code more readable, efficient

- `mutate()`: add new columns to data frame

- generally: good style to create a new object than clobbering the one that comes from an external source
- 여러 argument 동시에 줄 수 있음
- useful functions…

- `arrange()`: reorder rows in a data frame

- `sort()` will sort a vector but not a data frame

- to break ties: 2번째 argument 추가

-`summarize()`: collapses a data frame to a single row

- when used alone, collapses a data frame into a single row
- need to specify how we want to reduce an entire column of data into a single value
- `n()`: counts the number of rows
- use `group_by()` first to compare

3.2. A Grammar of Data Wrangling: With Multiple Tables

Relational Data Wrangling Using `dplyr`

-three families of verbs that work with two tables at a time:

- mutating joins: add new variables to one table from matching rows in another
- filtering joins: filter observations from one table based on whether or not they match an observation in the other table
- set operations: combine the observations in the data sets as if they were set elements

Types of Joins

-`inner_join()`: join data, retain only rows in both sets

-`left_join()`: row of the first table are always returned → NA's are inserted

-`semi_join()`: all rows in the first table that have a match in the second table

-`anti_join()`: all rows in the first table that do not have a match in the second table

Lecture 4. Statistical Foundation

4.1. Statistical Foundation

Data Science Workflow

-ultimate object in data science: extract meaningful information from data

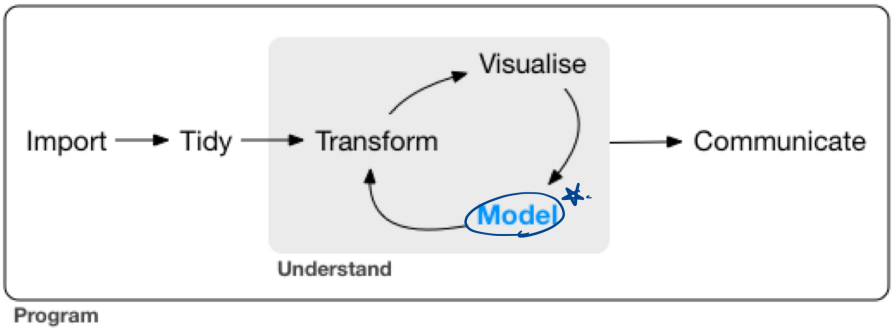
-data wrangling and visualization are just tools

- wrangling: to make data easier to interpret
- visualization: to connect our minds with the data, and to find patterns

-statistical methods

- quantify patterns and their strength
- find patterns that are too complex to be seen visually
- involves modeling

-workflow:



Exploratory Data Science (EDA)

-purpose

- to generate hypotheses

-both visualization and modeling are important tools in EDA

-typically involves many iterations of ‘transform - visualize - model’

Confirmatory Data Analysis

-purposes

- to confirm that the fitted model is appropriate
- to confirm the hypothesis is true

-predict accuracy vs. statistical inference?

-use visualization to graphically convey the conclusion

-data used in EDA are not used in confirmation generally

-statistical methodologies are both used in exploration and confirmation

- models
- (numerical) methods to fit the model
- assumptions and situations under which models can be used
- (probabilistic) methods to quantify the uncertainty in model fit
- quantification of the strength of evidence for/against hypotheses

Example: t-test

-hypothesis: the mean of X_i is $\mu = 0$

-t-test: reject the hypothesis if $t > 1.96$ or $t < -1.96$ where $t = \sqrt{n}\frac{mean(X)}{std(X)}$

→ more than the formula!

- model: observations are centered at μ(unknown), with certain amount of error
- fit: the center and amount of error are fitted by computing mean(X) and std(X)
- assumption: observations are numeric and follow a normal distribution
- uncertainty quantification: we are 95% confident that the true μ is in the interval $mean(X) \pm 1.96 \cdot std(X)/\sqrt{n}$
- strength of evidence: the smaller p-value, the stronger evidence against the hypothesis

Samples and Populations

-we’ve considered data as being fixed

- methodology assumes that the cases are drawn from a much larger set of potential cases
- view our sample of cases in the context of this population → imagine other samples that might have been drawn from the population

Example: Setting Travel Policy

-sample을 통해 계산한 q98 < population의 q98 → how to judge the result is good enough?

4.2. Sample Statistics

Sample Statistics

-a statistic: number that summarizes data

- mean of a variable, standard deviation, median, maximum, minimum
- maximum and minimum are not very useful

Sampling Distribution

-need to figure out the reliability of a sample statistic from the sample itself

-sampling distribution이 좁으면 좁을수록 good

-theory(CLT)에 의존 vs. bootstrap

-some standardized vocabulary

- sample size: the number of cases in the sample
- sampling distribution: the collection of the sample statistics from all of the trials (but exact number of trials is not important)
- shape: skewness, …
- standard error: standard deviation of the sampling distribution
- 95% confidence interval: another way of summarizing the sampling distribution
  → calculated from the mean and standard error of the sampling distribution

Sample Size n

-what size of sample n is needed to get a result with an acceptable reliability?

-sample size 커질수록 se 줄어듦

4.3. Bootstrap

Bootstrap: Approximating the Sampling Distribution

-bootstrap: statistical method that allows us to approximate the sampling distribution even without access to the population

-think of our sample itself as if it were the population!

-resampling: drawing a new sample from an existing sample → now: from our original sample

-when resampling, we do allow duplication (→ allows us to estimate the variability of the sample)

→ ”sample with replacement”

-결국, data-driven (non-parametric) way to estimate the *standard error* or *confidence interval*

Example: `nycflights13`

-500번 bootstrap한 결과 vs. 500개의 표본 cncnf

- mean은 다르지만 (≃ orig. sample의 mean) standard deviation은 꽤 비슷함

Bootstrap Distribution

-bootstrap distribution: the distribution of values in the bootstrap trials

-not exactly same as the sampling distribution, but for moderate to large sample sizes and sufficient number of bootstraps

- proven to approximate standard error or quantile of the sampling distribution

-CLT를 이용한 sampling dist도 결국 approximation: n=200일 때 normal dist인지 장담할 수 없음 → 실제 퍼포먼스 bootstrap이 더 좋음

-CLT는 mean에만 적용가능 vs. bootstrap은 (아무) 통계량이나 가능

-check the reliability of estimate using bootstrapping: confidence interval

1. $\hat{q_{98}}$(sample)$\pm 1.96SE$ (SE of bootstrap dist): normal dist를 따른다는 가정
2. bootstrap dist의 $[q_{2.5}, q_{97.5}]$: stable하지 않을수도 있음 → bootstrap 100,000번 하면… (bootsrap dist의 안정성/선명성에만 효과가 있음 - sampling dist처럼 mean 근처로 좁아지거나 하지 않음)

→ 1, 2는 보통 비슷함

4.4. Outliers

Identifying Extreme Events

-outlier: unusual or extreme events → 거기에도 무언가 의미가 있지 않을까?

-e.g. `nycflights13` example: carrier/month 와의 관계?

4.5. Statistical Models

Models

-model: an idealized representation of a system

- e.g. weather forecast: uses data collected in the past → make predictions about the future

-why do we build models?

1. to understand complex phenomena occurring in the world we live in
    - e.g. what factors play a role in today's weather?
    - care about creating simple and interpretable models
2. to make accurate predictions about unseen data
    - e.g. can we predict how much it will rain tomorrow?
    - care more about extremely accurate predictions, at the cost of having an uninterpretable model
    - sometimes called black-box models

→ most of the time, we try to strike a balance between interpretability and accuracy

Statistical Models

-used 'month' and 'carrier' to narrow down the situations in which the risk of an unacceptable flight delay is large

- *explaining* part of the variation in arrival delay

-statistical modeling provides a way to relate variables to one another

-quantitative response Y and p different predictors ($X_1, X_2, ..., X_p$)

- assume that there is some relationship between Y and $X = (X_1, X_2, ..., X_p)$
- $Y = f(X) + \epsilon$
    - f is some fixed, but unknown function
    - ϵ is a random error

Prediction and Inference

-two main reasons that we may wish to estimate f: prediction and inference

- can predict Y using $\hat{Y} = \hat{f}(X)$ → accuracy of $\hat{Y}$ is important
- for inference, we are more interested in the form of f
    - which predictors are associated?
    - what is the relationship?

- can relationship be summarized using a linear equation?

Reducible and Irreducible Error

-assume for a moment that both $\hat{f}$ and X are fixed → only variability comes from ϵ

-then,

$$\begin{aligned} E[(Y-\hat{Y})^2] &= E[(\hat{f}(X)+\epsilon-\hat{f}(X))^2)] \\ &= E[(f(X)-\hat{f}(X))^2] + E(\epsilon^2) \\ &= [f(X)-\hat{f}(X)]^2 + Var(\epsilon) \end{aligned}$$

- $[f(X)-\hat{f}(X)]^2$ is the reducible error (0 if $\hat{f}=f$)
- $Var(\epsilon)$ is the irreducible error → 줄이는 방법: 새로운 변수를 추가

-the focus is on techniques for estimating f with the aim of minimizing the reducible error

Parametric or Nonparametric Models

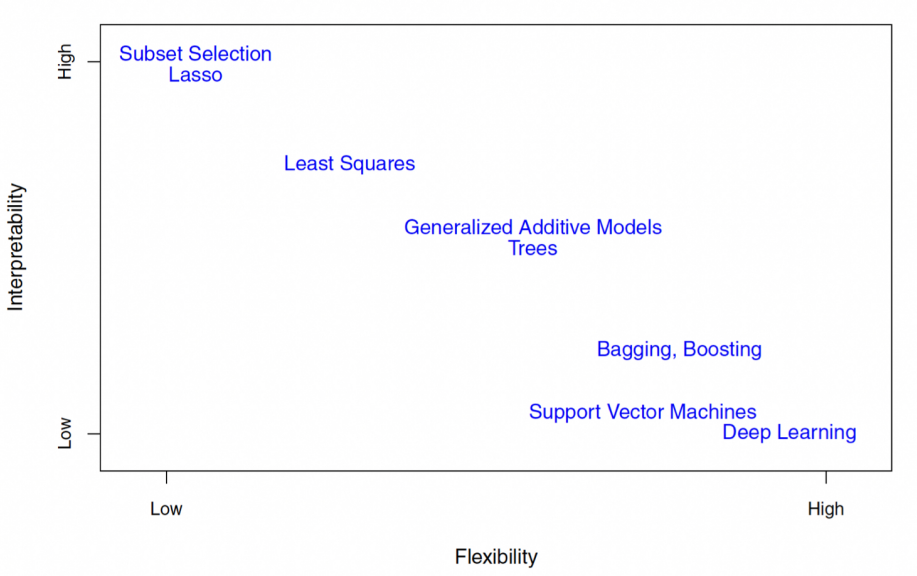
-when using a model f, we can use parameters $\theta=(\theta_1,...,\theta_p)$

-in such cases: use $f(X;\theta)$

-non-parametric models do not make explicit assumptions about the functional form of f (형태 주어지지 않음)

- e.g. k-nearest neighbors

Interpretability and Flexibility(Accuracy)



Example

-arrival delay depends on the hour?

1. use standard box-and-whisker plots
2. linear model (a kind of statistical model) → residual(line 기준 벗어나는 정도)이 normal distribution을 따르지 않음 (right-skewed)
3. how?

Lecture 5. Modeling and Linear Regression

5.1. Simple Linear Regression (SLR)

Introduction

-most common goal is statistics:

- “is the variable X associated with a variable Y, and if so, what is the relationship and can we use it to predict Y?”

-prediction of Y based on the values of other predictor variables is important both in statistics and data science

-simple approach: using linear regression

-supervised learning can be used

Prediction vs. Estimation

-subtle difference

- estimation: task of using data to determine model parameters
- prediction: task of using a model to predict outputs for unseen data
  - once we have estimates for our model's parameters, we can use it for prediction
  - non-parametric도 있으므로 prediction이 더 큰 개념

Simple Linear Regression

-linear regression can help us understand how values of a quantitative outcome/response are associated with values of a quantitative explanatory variable.

-applied in two ways:

- to generate predicted values
- to make inferences regarding associations in the dataset

-outcome = 종속변수 / predictor = 독립변수

-simple linear regression model for an outcome y as a function of a predictor x:

$$y_i = \beta_0 + \beta_1 x_i + \epsilon_i \text{ for } i = 1, ..., n$$

- n represents the number of observations in the data set
- β_0 = the population intercept coefficient
- β_1 = the true slope coefficient
- ϵ_i's = errors → assumed to be random noise with mean 0

-almost never know the true values of β_0 and β_1 → estimate them using data from sample

- `lm()` finds the 'best' coefficients $\hat{\beta}_0$ and $\hat{\beta}_1$ where the fitted values are given by $\hat{y}_i = \hat{\beta}_0 + \hat{\beta}_1 x_i$

-what is left over is captured by the residuals ($\hat{\epsilon}_i = y_i - \hat{y}_i$)

-model almost never fits perfectly

Example: RailTrail Data

-relationship between ridership(outcome) ~ [temperature, rainfall, cloud cover, day of the week]

-simple linear model: $volume_i = \beta_0 + \beta_1 hightemp_i + \epsilon_i$

- 잘 fitted 됐는지는 residual plot으로 확인
- $\hat{\beta}_0$: not interesting ($x = 0$ too low)
- $\hat{\beta}_1$: predicted increase in trail users for each additional degree in temp.

-best-fitting regression line: determined by a least squares criteria

- minimizes the sum of the squared residuals (= ϵ_i^2)
- the least squares regression line (defined by the values of $\hat{\beta}_0, \hat{\beta}_1$) is unique

-null model vs. SLR model → `lm` function 이용하면 구할 수 있음

- $\sum_{i=1}^{n}(y_i - \beta_0)^2$
  - $\hat{\beta}_0 = \bar{y}$ 일 때 최소
- $\sum_{i=1}^{n}(y_i - (\beta_0 + \beta_1 x_1))^2$ → 미분
  - $\hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \bar{x}$,
  - $\hat{\beta}_1 = \frac{\sum(y_i - \bar{y})(x_i - \bar{x})}{\sum(x_i - \bar{x})^2}$ 일 때 최소

-SLR is better(null model은 SLR에서 $\hat{\beta}_1 = 0$인 경우)→ how much better?

- residual sum, i.e. $r_i = \hat{\epsilon}_i = y_i - (\beta_0 + \beta_1 x_1)$의 제곱이 작으면..

- `lm` function 결과 해석

- `tidy(mod)`: statistic(t-value) = estimate / std.error
- `summary(mod)`: residual의 분포, R-squared 값

- `predict()`의 이용

- `mod$fitted.values[1:3]`과 `predict(mod)[1:3]`은 동일함: 기존 data
- `predict(mod, newdata)` → 새로운 $\hat{y}$ 값 출력
- `predict(mod, interval = "confidence")[1:3, ]` → fitted value의 confidence interval
  - $E[Y|X = x]$: x값 주어졌을 때 나오는 평균적인 pattern의 variability
- `predict(mod, interval = "prediction")[1:3, ]` → fitted value의 prediction interval
  - $Y|X = x$: 새로운 observation이 생겼을 때 어디에 위치하는지 (데이터가 안에 들어갈 확률이 95%)

Measuring the Strength of Fit

-R^2 is a common measure of goodness-of-fit for regression models

$$R^2 = 1 - \frac{SSE}{SST} = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y}_i)^2}$$

- SSE = 모델로 설명 못한 variability → 0이면 가장 좋음
- SST = 원래 데이터가 가지고 있는 variability

-R^2 is always between 0 and 1

- null model의 경우: SSE = SST

-a model with a larger R^2 is better

Least Squares

-SST is fixed(x와 관계없이) → smaller value of SSE indicates a good model

-actually, the least-square model finds the model with smallest SSE (by def.)

-mathematically, finding the best model = optimization problem: find $\hat{\beta}_0$, $\hat{\beta}_1$ such that

$$\sum_{i=1}^{n}(y_i - (\hat{\beta}_0 + \hat{\beta}_1 x_i))^2 \leq \sum_{i=1}^{n}(y_i - (\beta_0 + \beta_1 x_i))^2$$

- for all β_0, β_1 in the family

→ SSE의 최솟값을 찾는 것과 위 식의 $\hat{\beta}_0$, $\hat{\beta}_1$을 찾는 것(least squares method)은 동치

-historically, computational convenience is one reason for the widespread use of least squares in regression

- still computational speed is an important factor
- least squares are sensitive to outliers → sample size 커지면 ok

-many statistical methods or ‘machine learning’ algorithms are no unlike the SLR

-any statistical method optimizes a ‘goodness-of-fit measure’ over a family of models

- optimization comes down to formula
- often, algorithms are used to approximate the optimal member of the family: ‘gradient descent’
- ‘multiple’ families of models → computational efficiency becomes more important

Categorical Explanatory Variables

-`weekday` variable: binary

- $volume_i = \beta_0 + \beta_1 weekday_i + \epsilon_i$
- 그냥 regression 돌리면 β_1은 다른 카테고리로 넘어갈 때의 변화를 설명함

-`coef(lm(volume ~ weekday, data = RailTrail))`

- β_0: weekday = 0(weekend)일 때의 평균값
- $\beta_0 + \beta_1$: weekday = 1(weekday)일 때의 평균값 (0, 2로 코딩하면 1/2)
- β_0도 중요
- possible predicted values 더 증가할 경우?
  - 0, 1, 2로 mapping → not good (등간격으로 가정)
  - dummy variable 2개 → parameter 2개로 늘어남 → 더 flexible

-by default, the `lm` function will pick the alphabetically lowest value of the categorical predictor as the reference group and create indicators for other levels

5.2. Loss Functions

Loss Functions

-need some metric of how good or bad our predictions are

-a loss function characterizes the cost, error of fit resulting from a particular choice of model or model parameters

-looks like $L(y, \hat{y})$

- e.g. $L(y, \hat{y}) = (y - \hat{y})^2$

-affect accuracy and computational cost of estimation

L2 and L1 loss

-two popular choices: L1 loss (absolute loss) / L2 loss (squared loss)

- L2 loss: $L(y, \hat{y}) = (y - \hat{y})^2$ → SSE와 깊은 연관: 실은 SLR은 L2 loss를 최소화하는 것!
- L1 loss: $L(y, \hat{y}) = |y - \hat{y}|$

-L2 loss is widely used, but both are reasonable

- L2 loss produces a larger loss for a larger residual $e = (y - \hat{y})$
- loss의 max 정의 / 특정 값 이상에서는 L1 loss 사용

Empirical Risk (=Average Loss)

-for SLR model, $\hat{y} = \beta_0 + \beta_1$ → L2 loss: $L(y, \hat{y}) = (y - (\beta_0 + \beta_1 x))^2$

-want to find β_0, β_1 that minimized the total loss across all data points

- natural measure is of the average loss across all points

$$\hat{R}(\theta) = \hat{R}(\beta_0, \beta_1) = \frac{1}{n} \sum_i L(y_i, \hat{y}_i)$$

-the average loss on the **sample** tells us how well it fits the data (not the population!)

-for L2 loss, the average loss is the same as the mean squared error (MSE)

- MSE = $\frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$

-for L1 loss, the average loss is the same as the mean absolute error (MAE)

- MAE = $\frac{1}{n} \sum_{i=1}^{n} |y_i - \hat{y}_i|$

-the least square model is equivalent to minimizing the L2 loss function! (다른 loss를 쓰면 안 나올 수도)

-to find the best values, we set derivatives equal to zero

- need to solve the following estimating equations
  - $\frac{1}{n} \sum_i (y_i - \hat{y}_i) = 0$
  - $\frac{1}{n} \sum_i (y_i - \hat{y}_i) x_i = 0$
- 결과는 위 참조

Revisit the Constant (Null) Model

-when using a constant model, it seems that $\hat{\beta}_0 = \bar{y}$ is the natural estimator

- true when we use the L2 loss function
  - estimating equation: $\frac{1}{n} \sum_i (y_i - \beta_0) = 0$
  - solution: $\hat{\beta}_0 = \bar{y} \leftrightarrow \arg\min_{\beta_0} R(\beta_0) = \bar{y}$
- when we use L1 loss function?

- need to find $\hat{\beta}_0$ such that $\sum_{y_i<\beta_0} 1 = \sum_{y_i>\beta_0} 1$ → β_0 보다 작은 개수를 구한다는 것과 동일
- $\hat{\beta}_0 = \text{median}(y)$

$$\frac{d}{d\beta_0}R(\beta_0) = \frac{1}{n}\sum_i(\frac{d}{d\beta_0}|y_i - \beta_0|)$$
$$= \frac{1}{n}(\sum_{y_i<\beta_0}(-1) + \sum_{y_i<\beta_0}(+1)) = 0$$

Example: Gongcha Sales

-둘 다 (loss function 아님 주의) convex임은 증명 가능 (minimum is global)

- MSE: smooth → easy to minimize
- MAE: piecewise → hard to minimize

-Outliers?

- MSE: sensitive to outliers
- MAE: more robust to outliers

-Solution?

- MSE: unique solution as the sample mean
- MAE: not unique / sample size 커지면 구하기 어려움 (computational cost 증가)

→ outlier 많거나 sample size 작으면 MAE가 나을 수도

5.3. Multiple Regression

Multiple Regression

-natural extension of simple linear regression → multiple explanatory variables

-general form: $y = \beta_0 + \beta_1 x_1 + \cdots + \beta_p x_p + \epsilon$

- estimated coefficients are ‘conditional on’ the other variables
- each β_j reflects the predicted change in y associated with a one-unit increase in x_i, conditional upon the rest of the explanatory variables

-in practice:

- how can we quantify uncertainty of the estimated coefficients? → 가설검정, confidence interval 계산
- do all the predictors help to explain y? or is only a subset of the predictors useful? → model selection
- given a set of predictor values, what response value should we predict and how accurate is our prediction?

Inference for Regression

-what can we say about the population from our fitted model? → inference

-uncertainty quantification arises in two ways:

1. in prediction of 'y' from 'x'
2. in estimation of the coefficient, i.e. β_1 → 여기에 집중

-if do not know the population: use resampling methods like bootstrap

-for regression coefficient β_i's, the 95% confidence intervals can be constructed using normal approximation

$$\hat{\beta}_i \pm 1.96 \cdot SE(\hat{\beta}_i)$$

→ 2가지의 approximation: 1) $\hat{\beta} \sim \mathrm{N}(\beta, \sigma^2)$, 2) $SE(\hat{\beta}) \sim \sigma^2$

- bootstrap distribution을 이용해 2)를 approximate → 더 좋은 performance

How to Find a Better Model?

-can add many variables → R^2 is always improved

-but does not necessarily mean we have a better model

-principle of Occam's razor: all things being equal, a simpler model should be used in preference

-model selection:

- adjusted R^2
  - $R^2_{adj} = 1 - (1 - R^2)\frac{n-1}{n-p-1}$
  - p is the number of variables → give a penalty term for large p
- AIC(Akaike's Information Criteria): penalizes adding terms to a model
- BIC(Bayesian Information Criteria): similar to AIC, with a stronger penalty on p

- Mallow's C_p

→ performance 비슷, 주로 AIC를 기준으로 함

-how to find a model that minimizes AIC?

- all subset regression: search through all possible models
  - total of 2^p models that contain subsets of p predictors
  - computationally expensive
- stepwise regression
  - backward elimination: full model → drop variables
  - forward selection: constant model → add variables
  - mixed selection: combination of backward and forward selection

  → use `stepAIC` function
- penalized regression
  - AIC는 goodness-of-fit measure로 평가할 때 penalty / 애초에 parameter estimation 할 때 필요 없는 β → 0
  - model-fitting equation incorporates a constraint that penalizes the model for too many variables
  - applies the penalty by reducing coefficients
  - ridge regression: 목적함수 = L2 loss + L2 penalty
  - lasso regression: 목적함수 = L2 loss + L1 penalty

  → often called regularization (중간고사 이후에 다룰 예정)

Overfitting

-stepwise regression and all subsets regression are **in-sample** methods to assess and tune models

→ possibly subject to overfitting and may not perform as well when applied to new data

-to avoid this: use cross-validation (resampling method 중 하나)

-very important tool for more sophisticated types of models

- sample을 split → (k-1)개의 in-sample 모델을 하나의 out-sample에 적용 (k-fold)

Lecture 6. Regression and Resampling Methods

Introduction to Generalized Linear Models (GLM)

-so far: quantitative (or continuous outcomes)

- modeling a dichotomous outcome?
- outcome with several levels?
- count outcome?

-use monotonic *link function* f and consider

$$f(\theta_i) = \beta_0 + \beta_1 x_1 + \cdots + \beta_p x_p$$

- $y \sim D(\theta, \psi)$
  - θ_i is the center and ψ controls the scale or shape of the distribution

-y_i의 mean을 변환한 것이기 때문에 error term 없음 → conditional mean model 이라고도 함

6.1. Logistic Regression for Binary Outcome

Logistic Regression

-simple case: y is binary

- $y \sim Bernoulli(\theta)$
- link function f is the logit function

$$\text{D}$$

- $\theta = \frac{\exp(\beta_0+\beta_1 x_1+\cdots+\beta_p x_p)}{1+\exp(\beta_0+\beta_1 x_1+\cdots+\beta_p x_p)} = \frac{1}{1+\exp\{-(\beta_0+\beta_1 x_1+\cdots+\beta_p x_p)\}}$
- $0 < \theta < 1$

-`glm(has_diabetes ~ BMI + Age, family = "binomial")`

- significant? → p-value
- both predictors seems to be important

Predicted Probability

-predicted probability

$$\theta_i = P(y_i = 1) = \frac{\exp(\beta_0 + \beta_1 BMI_1 + \beta_2 Age_i)}{1 + \exp(\beta_0 + \beta_1 BMI_1 + \beta_2 Age_i)}$$

Understanding β Coefficients

-non-linear → change in the outcome variable is not a constant change in the predictor

-"odds ratios"

- comparing two odds: $\frac{p}{1-p} / \frac{q}{1-q}$
  - larger than 1 if $p > q$
- factor by which odds for the job change for someone who gains an MBA

-consider a logistic regression model with a single dichotomous predictor

$$\log(\frac{P(y_i = 1)}{1 - P(y_i = 1)}) = \beta_0 + \beta_1 x_i$$

- log odds that $y_i = 1$ when $x_i = 1$ is $\beta_0 + \beta_1$
- log odds that $y_i = 1$ when $x_i = 0$ is β_0
- denote $P(y_i = 1 \mid x_i = 1) = p$ and $P(y_i = 1 \mid x_i = 0) = q$, then

$$e^{\beta} = \frac{p}{1-p} / \frac{q}{1-q}$$

→ general interpretation for the β coefficients

- e.g. 발병 확률이 몇 % 증가하는가?

6.2. Generalized Linear Models for Count Outcomes

Modeling Count Data

-count data must be non-negative integers

Poisson Distribution

-used to model the probability of given number of events occurring in a fixed interval of time

-has a single parameter λ, known as the *rate*

- PMF: $P(X = k|\lambda) = \frac{e^{-\lambda}\lambda^k}{k!}$
- mean = variance = λ
- as λ increases, so too does the variance

Poisson Regression

-assume that each y_i is a Poisson random variable rate λ_i and

$$\log(\lambda_i) = \beta_0 + \beta_1 x_1 + \cdots + \beta_p x_p$$

- link function is the log function
- no optional parameter ψ in this case
- e^{β_j} is the multiplicative effect of an one unit increase in x_j
  - $\lambda_{x_j+1}|x_1, ..., x_{j-1}, x_{j+1}, ..., x_p = \lambda_{x_j}|x_1, ..., x_{j-1}, x_{j+1}, ..., x_p \cdot e^{\beta_j}$

Example: Biochemistry Ph.D.

-simple model: $\log(\lambda_i) = \beta_0 + \beta_1 \cdot x_1$

- $x_1 = 1$ if men and $x_1 = 0$ if women

```
glm(art ~ gender, family = "poisson")
```

- $\lambda_{men} = e^{\beta_0+\beta_1} = 1.881$
- $\lambda_{women} = e^{\beta_0} = 1.470$

→ exponent of gender coefficient e^{β_1} signifies the multiplicative increase to the average of articles for men relative to woman

(men produce articles on average $e^{\beta_1} = 1.280$ times more than woman)

Overdispersion

-overdispersed: if the variance of a sample is greater than would be expected according to a given theoretical model

-in count data: if variance of sample is much greater than its mean

→ using Poisson distribution could be a model mis-specification

Negative Binomial Distribution

-negative binomial distribution: distribution over non-negative integers

- probability distribution over the number of failures before r successes
- r번 성공 위해 던져야 하는 횟수 ↔ r번 성공 전에 나오는 뒷면의 횟수
- PMF: $P(x = k|r, \theta) = \binom{r+k-1}{k}\theta^r(1-\theta)^k$
- mean: $\mu = \frac{1-\theta}{\theta} \times r$
- $\theta = \frac{r}{r+\mu}$
- can parameterize the distribution by μ and r

-negative binomial distribution is equivalent as weighted sum of Poissons

-appropriate to use when the data can be seen as arising from a mixture of Poisson Distriutions

-assume that $y_i \sim \text{NegBinomial}(\mu_i, r)$

-then, $\log(\mu_i) = \beta_0 + \beta x_i$

- `glm.nb(art ~ gender)` vs. Poisson model
  - 회귀계수의 mean estimate는 같음
  - but std.error 커짐: overdispersed 설명

Zero-Inflated Poisson Distribution

-assume that each y_i is distributed as a Zero-Inflated Poisson mixture model:

$$y_i \sim \begin{cases} \text{Poisson}(\lambda_i) & \text{if } z_i = 0, \\ 0, & \text{if } z_i = 1 \end{cases}$$

- where rate λ_i and $\theta_i = P(z_i = 1)$ are functions of the predcitor x_i

Zero-Inflated Poisson Regression

-consider zero-inflated poisson regression, and y_i is from the mixture distribution, $\theta_i \cdot 1(y_i = 0) + (1 - \theta_i) \cdot \text{Poisson}(\lambda_i)$

-the θ_i and λ_i variables are the usual suspects

- $\log(\lambda_i) = \beta_0 + \beta_1 x_i$, and
- $\log(\frac{\theta_i}{1-\theta_i}) = \gamma_0 + \gamma_1 x_i$

-λ_i is modeled just as in ordinary Poisson regression

-θ_i is modeled in logistic regression

`zeroinfl(cigs ~ educ)`

- the Poisson model: $\log(\lambda_i) = 2.698 + 0.035x_1$
- the logistic model: $\log(\theta_i/(1-\theta_i)) = -0.563 + 0.084x_1$ (θ_i = non-smoker 일 prob.)
- when $z_i = 1$, i is a non-smoker → use logistic model
- for smokers, we use the Poisson model

6.3. Evaluating Models

Predictive Modeling

-$y = g(x) + \epsilon$

- in linear regression models: $g(x) = \beta_0 + \beta_1 x_1 + \cdots + \beta_p x_p$
- in GLM: $g(x) = f^{-1}(\beta_0 + \beta_1 x_1 + \cdots + \beta_p x_p)$ where f is a link function

-how can we decide which model is better? → bias and variance

Example: A Constant Model

-estimate how often a coin lands on head when flipped

- model A: select a random number between 0 and 1
- model B: select 0.7

-model A: on average, we will select 0.5 → zero bias, but lots of variance

-model B: will never be correct but not that terrible → zero variance, lots of bias(0.2)

→ bias-variance trade-off

Decomposition

-expected mean square of prediction:

$$E[(y - \hat{y}(x))^2] = E[\epsilon^2] + (g(x) - E[\hat{y}(x)])^2 + E[(E[\hat{y}(x)] - \hat{y}(x))^2]$$

-model risk = observation variance + (model bias)^2 + model variance

- observation variance: $\sigma^2 = E[\epsilon^2]$

    - y has a random noise ϵ that is irreducible error
    - measurement error or missing information acting like noise
- model bias: $(g(x) - E[\hat{y}(x)])^2$
    - difference between the expected predictions and the true $g(x)$
    - model may be too limited to find correct $g(x)$ → quadratic, cubic, …
- model variance: $E[(E[\hat{y}(x)] - \hat{y}(x))^2]$
    - fitted model is based on a random sample
    - sample could have come out differently

Improve Each Component

-observation variance: beyond control

-model variance is often due to overfitting

- small differences in random samples → large differences in the fitted model
- reduce model complexity (avoid to fit the noise!)

-model bias is often due to underfitting (or lack of domain knowledge)

- increase model complexity

Bias-Variance Tradeoff

-as model complexity(e.g. number of features) increases,

- model bias decreases
- but model variance increases

Training Error

-overfitting is an issue we want to avoid

-when computing model risk → training error $\simeq$ bias

-model fitting 할 때와 다른 sample 사용해야 함

Holdout Method

-a.k.a. validation set approach

-hold out some units from sample → compute validation error

-3차식 이후로는 training error의 급격한 감소 없음 → 가장 작은 validation error

-AIC: gives a penalty term on p → 3차식이 best model

-no general guideline for this split ratio: 기본적으로 80/2 → sample size 커지면 test set 비율 작아짐

-test set을 잘못 선택했을 수도

Cross-Validation

-build model on the data in X_1 → run the data in X_2 → reverse

-if first model is overfit, it will not likely perform as well on the second set of data

-if we decide to use non-overlapping chunks of 20% of data, there are 5 possible chunks of data we could use as our validation set

- first fold = validation set, method is fit on the remaining k-1 folds
- mean squared error computed on the observations in the first fold
- repeat k times

-k-fold CV estimate: $CV_{(k)} = \frac{1}{k}\sum_{i=1}^{k} MSE_i$

-use `cv.glm()`

Leave-One-Out Cross Validation(LOOCV)

-always yield the same results

-not to over-estimate the test error rate

-less bias than the holdout approach

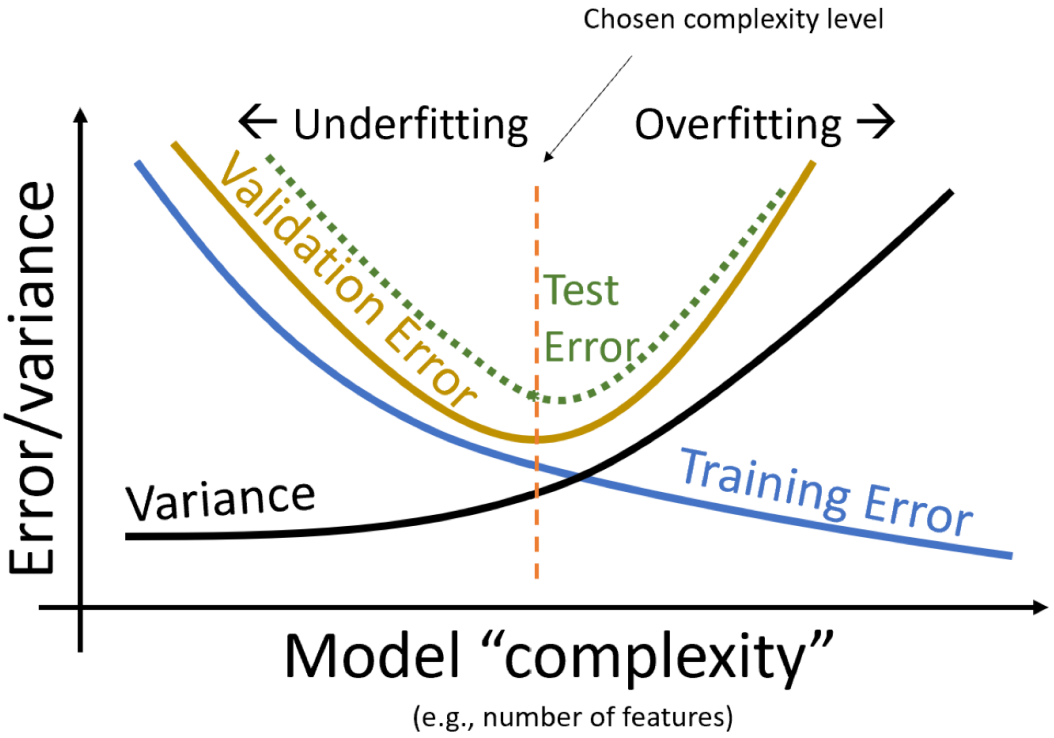
-but computationally expensive (linear regression인 경우 closed form solution 있음)

Why are Test Sets Needed?

-validation error < test error!

- cross-validation 과정 통해 training set 에서도 들어감
- validation error (sample 안에서) 계산할 때 운이 좋았을 수도

-test error의 최솟값과 validation error의 최솟값이 다를 수도 있음


Chosen complexity level
← Underfitting
Overfitting →
Validation Error
Test
Error
Training Error
Variance
Error/variance
Model “complexity”
(e.g., number of features)