R Journal Entry Analysis

HHH

2024 2 20

# Step 1. 테스트 개요 및 목적

재무제표에 대한 회계감사의 일환으로 감사대상기간 동안 발생한 모든 전표 데이터에 대한 무결성 및 비경상적인 거래가 존재하는지를 검증

1. 계정명(FSLI) : 전체 계정
2. 기준일 (Coverage date) : 2017-01-01 ~ 2017-12-31
3. 테스트되는 경영자의 주장 (Assertion) : 완전성 (C),정확성 (A),기간귀속구분 (CO),실재성 (E/O),권리 (R),공시(PD),평가(V)

# Step 2. Test 대상 모집단

1. 전체 전표를 포함하는 분개장.
2. 모집단의 완전성 확인 : Step 5. Test 결과의 A01, A02, A03 참조

# Step 3. 오류의 정의

재무보고 프로세스 관련, 부적절하거나 비경상적인 분개 및 수정사항

# Step 4. Test 방법

* A01 . Data Integrity 검증 - 데이터 유효성을 검증하고, record에 대한 이해를 위한 절차

전표 데이터의 회계기간이 당해년도에 포함되는지 여부 검토  
전표 주요 필드값의 누락 여부 검토를 통한 data integrity 검토

* A02. 전표번호 별 차대변 일치검증  
  전표번호 별 차변금액과 대변금액이 일치하는지 확인하여 전표 데이터의 완전성을 검토  
  차변금액과 대변금액이 일치하지 않을 경우 해당 전표를 추출하여 회사 측과 확인
* A03. 시산표 Reconciliation 검증(Trial Balance Rollforward Test)  
  기초 F/S잔액에 수령한 모든 전표의 계정과목 별 합계금액을 반영하여 도출한 기말 F/S 잔액과 회사 제시 F/S와의 일치 여부 검토
* B01. 매출의 상대계정분석(매출과 연관성이 낮은 계정이 포함된 비정상적인 거래)  
  매출에 대한 상대계정분석 결과 비정상적으로 처리된 회계처리가 있는지 검토하고 해당 전표 중 특정 금액 이상인 건을 추출하여 검토

# Step 5. Test 결과

## Test 준비

### readxl, lubridate, skimr, tidyverse 패키지를 불러온다.

## ── Attaching core tidyverse packages ──────────────────────── tidyverse 2.0.0 ──  
## ✔ dplyr 1.1.3 ✔ readr 2.1.4  
## ✔ forcats 1.0.0 ✔ stringr 1.5.0  
## ✔ ggplot2 3.4.4 ✔ tibble 3.2.1  
## ✔ lubridate 1.9.3 ✔ tidyr 1.3.0  
## ✔ purrr 1.0.2   
## ── Conflicts ────────────────────────────────────────── tidyverse\_conflicts() ──  
## ✖ dplyr::filter() masks stats::filter()  
## ✖ dplyr::lag() masks stats::lag()  
## ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors

### load Journal Entry data

* Journal Entry data 객체 load 하고 확인

je\_raw <- read\_csv('je\_utf\_colname.txt')

## Rows: 309043 Columns: 22  
## ── Column specification ────────────────────────────────────────────────────────  
## Delimiter: ","  
## chr (9): 계정과목명, 거래처[코드], 결의사원[코드], 유형번호[유형명], 증빙구분[구분명], INSERT\_ID, 입력일자, M...  
## dbl (11): 전표일자, 구 분[기표][번호], 전표번호, LN\_SQ, DRCR\_FG, 계정코드, 구 분[결의][년/월/일], 구 분...  
## num (2): 차변금액, 대변금액  
##   
## ℹ Use `spec()` to retrieve the full column specification for this data.  
## ℹ Specify the column types or set `show\_col\_types = FALSE` to quiet this message.

je\_raw

## # A tibble: 309,043 × 22  
## 전표일자 `구 분[기표][번호]` 전표번호 LN\_SQ DRCR\_FG 계정코드 계정과목명   
## <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <chr>   
## 1 20170101 1 201701011 1 3 12002 매출채권미수…  
## 2 20170101 1 201701011 2 4 10800 외상매출금   
## 3 20170101 1 201701011 3 4 10800 외상매출금   
## 4 20170101 46 2017010146 1 3 10800 외상매출금   
## 5 20170101 46 2017010146 2 4 40401 제품매출   
## 6 20170101 46 2017010146 3 4 25500 부가세예수금   
## 7 20170101 47 2017010147 1 3 10800 외상매출금   
## 8 20170101 47 2017010147 2 4 40401 제품매출   
## 9 20170101 47 2017010147 3 4 25500 부가세예수금   
## 10 20170101 48 2017010148 1 3 10800 외상매출금   
## # ℹ 309,033 more rows  
## # ℹ 15 more variables: 차변금액 <dbl>, 대변금액 <dbl>, `거래처[코드]` <chr>,  
## # `구 분[결의][년/월/일]` <dbl>, `구 분[결의][번호]` <dbl>,  
## # `결의부서[코드]` <dbl>, `결의사원[코드]` <chr>, `PJT[코드]` <dbl>,  
## # `유형번호[번호]` <dbl>, `유형번호[유형명]` <chr>, `증빙구분[구분명]` <chr>,  
## # INSERT\_ID <chr>, 입력일자 <chr>, MODIFY\_ID <chr>, MODIFY\_DT <chr>

### 프로그램에 맞추어 열 이름 변경

* change column name

je\_tbl <- rename(je\_raw, JEDATE = 전표일자,  
 JENO = 전표번호,  
 DR = 차변금액,  
 CR = 대변금액,  
 ACCTCD = 계정코드,  
 ACCT\_NM = 계정과목명)  
  
je\_tbl <- je\_tbl |>   
 mutate(DR = ifelse(is.na(DR), 0, DR),  
 CR = ifelse(is.na(CR), 0, CR),  
 JEDATE = ymd(JEDATE)  
)  
  
je\_tbl

## # A tibble: 309,043 × 22  
## JEDATE `구 분[기표][번호]` JENO LN\_SQ DRCR\_FG ACCTCD ACCT\_NM DR  
## <date> <dbl> <dbl> <dbl> <dbl> <dbl> <chr> <dbl>  
## 1 2017-01-01 1 201701011 1 3 12002 매출채… 132000  
## 2 2017-01-01 1 201701011 2 4 10800 외상매… 0  
## 3 2017-01-01 1 201701011 3 4 10800 외상매… 0  
## 4 2017-01-01 46 2017010146 1 3 10800 외상매… 332090  
## 5 2017-01-01 46 2017010146 2 4 40401 제품매… 0  
## 6 2017-01-01 46 2017010146 3 4 25500 부가세… 0  
## 7 2017-01-01 47 2017010147 1 3 10800 외상매… 216480  
## 8 2017-01-01 47 2017010147 2 4 40401 제품매… 0  
## 9 2017-01-01 47 2017010147 3 4 25500 부가세… 0  
## 10 2017-01-01 48 2017010148 1 3 10800 외상매… 66000  
## # ℹ 309,033 more rows  
## # ℹ 14 more variables: CR <dbl>, `거래처[코드]` <chr>,  
## # `구 분[결의][년/월/일]` <dbl>, `구 분[결의][번호]` <dbl>,  
## # `결의부서[코드]` <dbl>, `결의사원[코드]` <chr>, `PJT[코드]` <dbl>,  
## # `유형번호[번호]` <dbl>, `유형번호[유형명]` <chr>, `증빙구분[구분명]` <chr>,  
## # INSERT\_ID <chr>, 입력일자 <chr>, MODIFY\_ID <chr>, MODIFY\_DT <chr>

sum(je\_tbl$DR) == sum(je\_tbl$CR)

## [1] TRUE

## A01 Data Integrity Test

* Data skim 확인

skim(je\_tbl)

Data summary

|  |  |
| --- | --- |
| Name | je\_tbl |
| Number of rows | 309043 |
| Number of columns | 22 |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |  |
| Column type frequency: |  |
| character | 9 |
| Date | 1 |
| numeric | 12 |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |  |
| Group variables | None |

**Variable type: character**

| skim\_variable | n\_missing | complete\_rate | min | max | empty | n\_unique | whitespace |
| --- | --- | --- | --- | --- | --- | --- | --- |
| ACCT\_NM | 0 | 1.00 | 2 | 16 | 0 | 211 | 0 |
| 거래처[코드] | 2172 | 0.99 | 5 | 5 | 0 | 5909 | 0 |
| 결의사원[코드] | 0 | 1.00 | 10 | 10 | 0 | 18 | 0 |
| 유형번호[유형명] | 0 | 1.00 | 4 | 4 | 0 | 5 | 0 |
| 증빙구분[구분명] | 222980 | 0.28 | 2 | 12 | 0 | 9 | 0 |
| INSERT\_ID | 178226 | 0.42 | 10 | 10 | 0 | 17 | 0 |
| 입력일자 | 178226 | 0.42 | 21 | 22 | 0 | 20885 | 0 |
| MODIFY\_ID | 772 | 1.00 | 10 | 10 | 0 | 10 | 0 |
| MODIFY\_DT | 772 | 1.00 | 21 | 22 | 0 | 13253 | 0 |

**Variable type: Date**

| skim\_variable | n\_missing | complete\_rate | min | max | median | n\_unique |
| --- | --- | --- | --- | --- | --- | --- |
| JEDATE | 0 | 1 | 2017-01-01 | 2017-12-31 | 2017-07-10 | 326 |

**Variable type: numeric**

| skim\_variable | n\_missing | complete\_rate | mean | sd | p0 | p25 | p50 | p75 | p100 | hist |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 구 분[기표][번호] | 0 | 1.00 | 746.84 | 896.88 | 1 | 57 | 321 | 1135 | 3236 | ▇▂▁▁▁ |
| JENO | 0 | 1.00 | 65207179028.64 | 85984640164.26 | 201701011 | 2017092610 | 20170616160 | 201702221968 | 201712311321 | ▇▁▁▁▃ |
| LN\_SQ | 0 | 1.00 | 67.61 | 190.65 | 1 | 1 | 2 | 4 | 997 | ▇▁▁▁▁ |
| DRCR\_FG | 0 | 1.00 | 3.67 | 0.48 | 1 | 3 | 4 | 4 | 4 | ▁▁▁▃▇ |
| ACCTCD | 0 | 1.00 | 24348.93 | 15524.77 | 10100 | 10800 | 25301 | 40401 | 99800 | ▇▂▁▁▁ |
| DR | 0 | 1.00 | 939381.78 | 30948687.50 | -1000000000 | 0 | 0 | 55000 | 4483053872 | ▇▁▁▁▁ |
| CR | 0 | 1.00 | 939381.78 | 29883464.98 | -1777760000 | 0 | 10000 | 70000 | 4483053872 | ▁▇▁▁▁ |
| 구 분[결의][년/월/일] | 0 | 1.00 | 20170680.32 | 345.45 | 20170101 | 20170406 | 20170710 | 20171011 | 20171231 | ▇▆▆▆▇ |
| 구 분[결의][번호] | 0 | 1.00 | 740.23 | 896.64 | 1 | 50 | 314 | 1111 | 3225 | ▇▂▁▁▁ |
| 결의부서[코드] | 0 | 1.00 | 1936.41 | 844.57 | 1400 | 1400 | 1400 | 2900 | 4120 | ▇▁▁▂▁ |
| PJT[코드] | 78366 | 0.75 | 3150.05 | 542.66 | 1100 | 3100 | 3100 | 3200 | 6100 | ▁▇▃▁▁ |
| 유형번호[번호] | 0 | 1.00 | 2.92 | 0.85 | 1 | 3 | 3 | 3 | 8 | ▃▇▃▁▁ |

## A02 전표번호 별 차대변 일치검증

* 전표번호, 차변금액, 대변금액 열을 선택(select)한다. 선택한 세 개의 열에 Null 값 (NA)이 있으면 0으로 모두 바꾼다.(mutate\_all)  
  전표번호별로 묶어서(group\_by) 전표번호별 차변의 합계를 DR\_sum의 열 이름으로 전표번호별 대변의 합계를 CR\_sum의 열 이름으로 정리한다.(summarise)  
  전표번호별 차변 금액과 대변 금액 합계를 각각 차이를 계산하여 Differ라는 열 이름으로 추가(mutate)한다.
* 전표번호별 차변 금액 합계와 대변 금액 합계의 차이 Differ열의 전체 합계를 계산한다.
* 계산한 차이가 0이 아닌 행을 나타낸다.

A02 <- je\_tbl |>   
 select(JENO, DR, CR) |>   
 mutate\_all(~replace(., is.na(.), 0)) |>   
 group\_by(JENO) |>   
 summarise(DR\_sum=sum(DR),  
 CR\_sum=sum(CR)) |>   
 mutate(Differ= DR\_sum - CR\_sum)  
# print(A02)  
print(A02)

## # A tibble: 80,612 × 4  
## JENO DR\_sum CR\_sum Differ  
## <dbl> <dbl> <dbl> <dbl>  
## 1 201701011 132000 132000 0  
## 2 201701021 334400 334400 0  
## 3 201701022 708400 708400 0  
## 4 201701023 31900 31900 0  
## 5 201701024 145500 145500 0  
## 6 201701025 786500 786500 0  
## 7 201701026 33000 33000 0  
## 8 201701027 13838 13838 0  
## 9 201701028 148500 148500 0  
## 10 201701029 571000 571000 0  
## # ℹ 80,602 more rows

A02 |> filter(Differ != 0)

## # A tibble: 0 × 4  
## # ℹ 4 variables: JENO <dbl>, DR\_sum <dbl>, CR\_sum <dbl>, Differ <dbl>

## A03 시산표 Reconciliation 검증(Trial Balance Rollforward Test)

### 시산표 data 준비.

* 당기 시산표 불러오기
* 전기 시산표 불러오기
* 당기시산표 columns별 결측값 확인
* 전기시산표 columns별 결측값 확인

cytb\_raw <- read\_excel('CYTB.xlsx')  
pytb\_raw <- read\_excel('PYTB.xlsx')  
colSums(is.na(cytb\_raw))

## 차 변[잔 액] DRSUM 차 변[당월계] ACCTCD   
## 0 0 0 28   
## 계정과목 대 변[당월계] CRSUM 대 변[잔 액]   
## 0 0 0 0

colSums(is.na(pytb\_raw))

## 차 변[잔 액] DRSUM 차 변[당월계] ACCTCD   
## 0 0 0 28   
## 계정과목 대 변[당월계] CRSUM 대 변[잔 액]   
## 0 0 0 0

* 결측값(Null, NA)이 ACCTCD 열에 있었으며 결측값 때문에 join에 어려움이 발생할 수 있으므로 결측을 없애고 시작한다.
* 전기 시산표에서도 결측을 없앤다.
* 결측을 없애고 나서 View(CYTB) 기능으로 읽어들인 시산표를 보니 99행이 시산표 중 재무상태표의 마지막 행이다. 시산표의 재무상태표 부분을 나누어 새로운 객체를 생성하기 위하여 1행 부터 99행까지를 썰어낸다.(slice)
* 시산표의 손익계산서 부분을 뜯어내어 새로운 객체를 생성하기 위하여 100행부터 끝까지 썰어낸다.
* 시산표에서 썰어낸(slice) 당기재무상태표 하단을 확인한다. 재무상태표 끝단이 썰렸는지 확인.
* 시산표에서 썰어낸(slice) 당기손익계산서 상단을 확인한다. 손익계산서 시작점부터 썰렸는지 확인.
* 전기시산표에서 썰어낸 전기재무상태표 하단을 확인한다. 재무상태표 끝단이 썰렸는지 확인.
* 전기 시산표의 재무상태표 부분 slice
* 당기시산표에 전기 시산표의 재무상태표 부분을 계정과목별로 맞춰서 붙인다(join). 이 때 당기시산표에 있는 계정과 전기시산표에 있는 계정이 모두 포함되도록 붙여야 한다. 전기에 변동이 있었으나 당기에 변동이 없었던 항목이 누락되지 않아야 하기 때문이다.
* 시산표의 손익계산서 부분은 당기에 발생한 잔액만 있으면 된다. 재무상태표처럼 기초잔액이라는 개념이 없다.
* 재무상태표의 잔액에서 기초잔액을 제거한 결과와 시산표의 손익계산서 부분의 잔액을 합하여 전표에서 당기에 변동이 있었던 금액을 하나로 붙인다.

cytb\_tbl <- drop\_na(cytb\_raw, ACCTCD)  
pytb\_tbl <- drop\_na(pytb\_raw, ACCTCD)  
   
CYTB\_FP <- cytb\_tbl |>   
 slice(1:99)  
CYTB\_PL <- cytb\_tbl |>   
 slice(100:n())  
PYTB\_FP <- pytb\_tbl |>   
 slice(1:103)  
   
tail(CYTB\_FP)

## # A tibble: 6 × 8  
## `차 변[잔 액]` DRSUM `차 변[당월계]` ACCTCD 계정과목 `대 변[당월계]`  
## <dbl> <dbl> <dbl> <chr> <chr> <dbl>  
## 1 0 0 0 38300 자 기… 0  
## 2 56201942 3.09e7 72053772 38900 매도가… 15851830  
## 3 0 0 0 35100 이 익 … 0  
## 4 0 0 0 37200 부의지… 0  
## 5 0 3.12e7 31216627 37400 보험수… 36377018  
## 6 0 1.32e9 0 37500 전기이… 0  
## # ℹ 2 more variables: CRSUM <dbl>, `대 변[잔 액]` <dbl>

head(CYTB\_PL)

## # A tibble: 6 × 8  
## `차 변[잔 액]` DRSUM `차 변[당월계]` ACCTCD 계정과목 `대 변[당월계]`  
## <dbl> <dbl> <dbl> <chr> <chr> <dbl>  
## 1 0 0 0 40100 상 품… 468556458  
## 2 0 0 0 40401 제 품… 3765154556  
## 3 0 0 0 40700 교 육… 671429789  
## 4 0 0 0 41100 임 대… 70591818  
## 5 0 0 0 41200 기 타… 90910  
## 6 2790687018 2.79e9 626758061 45100 상 품 … 0  
## # ℹ 2 more variables: CRSUM <dbl>, `대 변[잔 액]` <dbl>

tail(PYTB\_FP)

## # A tibble: 6 × 8  
## `차 변[잔 액]` DRSUM `차 변[당월계]` ACCTCD 계정과목 `대 변[당월계]`  
## <dbl> <dbl> <dbl> <chr> <chr> <dbl>  
## 1 0 -7.14e6 0 38800 매도가… 0  
## 2 145999555 1.87e8 187178917 38900 매도가… 41179362  
## 3 0 0 0 35100 이 익 … 0  
## 4 0 0 0 37200 부의지… 0  
## 5 0 1.50e7 15037512 37400 보험수… 3308252  
## 6 0 0 0 37500 전기이… 0  
## # ℹ 2 more variables: CRSUM <dbl>, `대 변[잔 액]` <dbl>

CYTB\_FP\_move <- full\_join(CYTB\_FP, PYTB\_FP, by='ACCTCD') |>   
 mutate\_all(~replace(., is.na(.), 0)) |>   
 mutate(move = (DRSUM.x - CRSUM.x) -(DRSUM.y - CRSUM.y)) |>   
 select(ACCTCD, move)  
  
CYTB\_PL\_move <- CYTB\_PL |>   
 mutate(move = (DRSUM - CRSUM)) |>   
 select(ACCTCD, move)  
  
CYTB\_move <- bind\_rows(CYTB\_FP\_move, CYTB\_PL\_move)

* 전표에서 계정과목별 차변금액과 대변금액 합계를 계산하여 준비한다.
* 시산표에서 준비한 당기 변동금액(전표에서 계산한 계정과목별 당기 변동 금액)을 전표에서 계산한 차변금액과 대변 금액의 합계에 붙인다.
* 차이가 0보다 큰 항목 갯수 확인.
* A03 객체에서 차이가 0이 아닌 항목을 찾아 나타낸다. 이익잉여금 항목 하나만 나타나면 이 외의 계정은 모두 일치한다.

A03 <- je\_tbl |>   
 select(ACCTCD, DR, CR) |>   
 mutate\_all(~replace(., is.na(.), 0)) |>   
 group\_by(ACCTCD) |>   
 summarise(DR\_sum=sum(DR),  
 CR\_sum=sum(CR)) |>   
 ungroup() |>  
 # join 할때 자료형 일치  
 mutate(ACCTCD = as.character(ACCTCD))  
  
A03 <- left\_join(A03, CYTB\_move, by = 'ACCTCD')  
A03 <- A03 |> mutate\_all(~replace(.,is.na(.), 0)) |>   
 mutate(Differ = (DR\_sum - CR\_sum - move))  
  
ac\_name <- je\_tbl |>   
 distinct(ACCTCD, ACCT\_NM) |>   
 mutate(ACCTCD=as.character(ACCTCD))  
  
A03 <- left\_join(A03, ac\_name, by='ACCTCD')  
  
print(A03)

## # A tibble: 211 × 6  
## ACCTCD DR\_sum CR\_sum move Differ ACCT\_NM   
## <chr> <dbl> <dbl> <dbl> <dbl> <chr>   
## 1 10100 125649815 124027515 1622300 0 현금   
## 2 10200 1735072699 1735072699 0 0 당좌예금   
## 3 10301 90863358102 90172042718 691315384 0 보통예금   
## 4 10302 785906508 439897964 346008544 0 외화예금   
## 5 10501 7800000000 8400000000 -600000000 0 정기예금   
## 6 10502 125175743 347606170 -222430427 0 정기적금   
## 7 10600 -54781102 0 -54781102 0 단기금융상품   
## 8 10800 29594408190 29451728317 142679873 0 외상매출금   
## 9 10900 101829493 -230485439 332314932 0 대손충당금(외)  
## 10 11000 45660000 41745000 3915000 0 받을어음   
## # ℹ 201 more rows

A03 |> filter(Differ != 0)

## # A tibble: 1 × 6  
## ACCTCD DR\_sum CR\_sum move Differ ACCT\_NM   
## <chr> <dbl> <dbl> <dbl> <dbl> <chr>   
## 1 37500 1320667762 0 -2957850338 4278518100 전기이월이익잉여금

### Lead Schedule xlsx 생성

amt\_sum <- je\_tbl |>   
 mutate\_all(~replace(., is.na(.), 0)) |>   
 summarise(  
 DR\_sum = sum(DR),  
 CR\_sum = sum(CR),  
 .by = ACCT\_NM  
 ) |> arrange(ACCT\_NM)  
  
amt\_sum |> write\_xlsx("lead\_s.xlsx")

## B01 매출의 상대계정분석

* 상대계정분석에서 기준이 되는 본계정코드를 설정한다.
* 전표 데이터에서 본계정이 포함된 전표번호를 필터링(filter)한다. (행 선택)
* 전표 데이터에서 전표번호와 계정코드 열을 선택(select)한다. (열 선택)
* 선택한 전표번호와 계정코드에서 본계정이 포함된 전표번호만 남겨서 B09 변수로 지정한다.
* 본계정과 상대계정의 전표 갯수를 세어 B09 변수로 지정한다.

Corr\_Acc = '40401'  
B09\_main <- je\_tbl |> filter(ACCTCD == Corr\_Acc) |>   
 select(JENO, ACCTCD, ACCT\_NM)  
B09\_Corr <- je\_tbl |>   
 select(JENO, ACCTCD)  
B09 <- semi\_join(B09\_Corr, B09\_main, by = 'JENO')  
B09 <- B09 |> filter(!is.na(ACCTCD)) |>   
 count(ACCTCD)

계정과목명 확인

* 계정코드와 계정과목명 열을 선택한다. 중복되는 계정과목명을 삭제하고 하나씩만 남긴다.
* 계정코드에 맞는 계정과목명을 B09 에 붙인다.

acct\_nm <- je\_tbl |>   
 select(ACCTCD, ACCT\_NM) |>   
 distinct()  
  
B09 <- B09 |> left\_join(acct\_nm, by='ACCTCD')  
  
print(B09)

## # A tibble: 8 × 3  
## ACCTCD n ACCT\_NM   
## <dbl> <int> <chr>   
## 1 10800 57348 외상매출금   
## 2 11300 66 진행률미수금   
## 3 25500 57216 부가세예수금   
## 4 25900 75 선수금   
## 5 26900 2 프로젝트손실충당부채   
## 6 40100 44 상품매출   
## 7 40401 57419 제품매출   
## 8 54100 1 프로젝트손실전입액(제)

# Conclusion

전표 분석 결과 Step3에서 정의한 오류 사항이 발견되지 아니함.

# Analysis

## 벤포드의 법칙

* 벤포드 분석

if(!require(benford.analysis)){install.packages("benford.analysis");library(benford.analysis)}

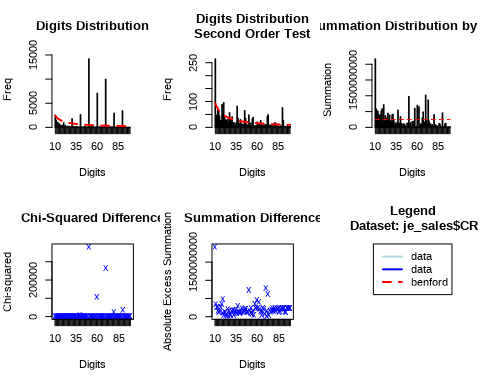
## Loading required package: benford.analysis

je\_sales <- je\_tbl |>   
 filter(ACCTCD == '40401')  
  
je\_sales$CR |> benford()

##   
## Benford object:  
##   
## Data: je\_sales$CR   
## Number of observations used = 57137   
## Number of obs. for second order = 2236   
## First digits analysed = 2  
##   
## Mantissa:   
##   
## Statistic Value  
## Mean 0.643  
## Var 0.073  
## Ex.Kurtosis 0.242  
## Skewness -1.179  
##   
##   
## The 5 largest deviations:   
##   
## digits absolute.diff  
## 1 50 13707.61  
## 2 70 9635.02  
## 3 60 6653.84  
## 4 90 3187.81  
## 5 80 2639.74  
##   
## Stats:  
##   
## Pearson's Chi-squared test  
##   
## data: je\_sales$CR  
## X-squared = 853103, df = 89, p-value < 0.00000000000000022  
##   
##   
## Mantissa Arc Test  
##   
## data: je\_sales$CR  
## L2 = 0.28011, df = 2, p-value < 0.00000000000000022  
##   
## Mean Absolute Deviation (MAD): 0.01517506  
## MAD Conformity - Nigrini (2012): Nonconformity  
## Distortion Factor: 30.68761  
##   
## Remember: Real data will never conform perfectly to Benford's Law. You should not focus on p-values!

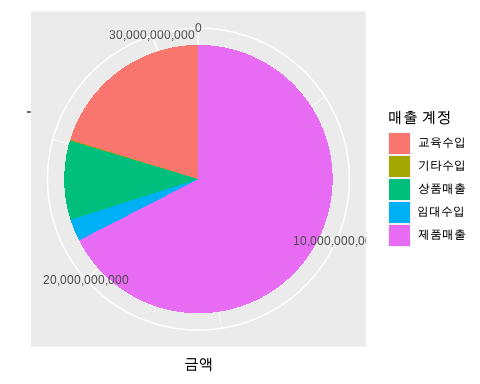
* 벤포드 분석 결과 Plot 확인

bl <- je\_sales$CR |> benford()  
plot(bl)



## 매출액 계정별 비중 확인

CYTB\_PL |>   
 slice(1:5) |>   
 select(ACCTCD) |>   
 left\_join(ac\_name, by = 'ACCTCD') |>   
 left\_join(amt\_sum, by = 'ACCT\_NM') |>   
 ggplot(aes(x="", y=CR\_sum, fill=ACCT\_NM)) +  
 geom\_bar(stat="identity", width=1) +  
 coord\_polar("y", start=0)+  
 scale\_y\_continuous(labels = scales::comma)+  
 xlab("") + ylab("금액") + labs(fill="매출 계정")



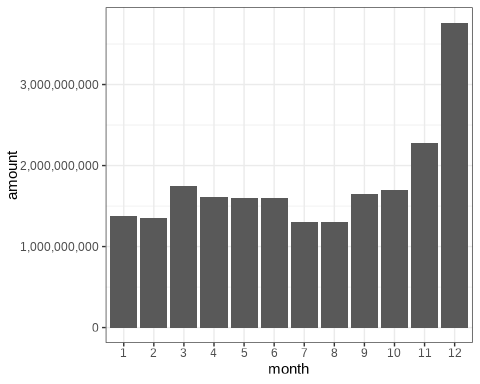
매출액 계정 중 제품매출의 비중이 가장 크다.

## 매출 월별 추세 분석

* 제품매출 월별 추세 분석

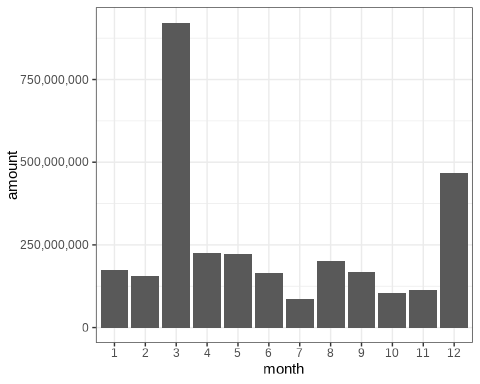
acc = '40401'  
  
trend <- \(acc){  
 je\_tbl |>   
 filter(ACCTCD %in% acc) |>   
 mutate(month = as.factor(month(JEDATE))) |>   
 summarise(  
 amount = sum(CR),  
 .by = month  
 ) |> ggplot(aes(x=month, y=amount))+  
 geom\_col()+  
 scale\_y\_continuous(labels = scales::comma)+  
 theme\_bw()  
}

trend(acc)



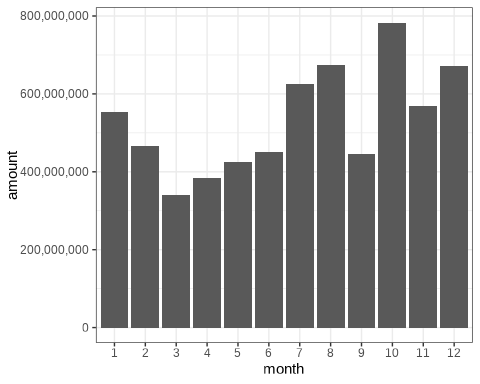
* 상품매출 월별 추세 분석

acc <- "40100"  
trend(acc)



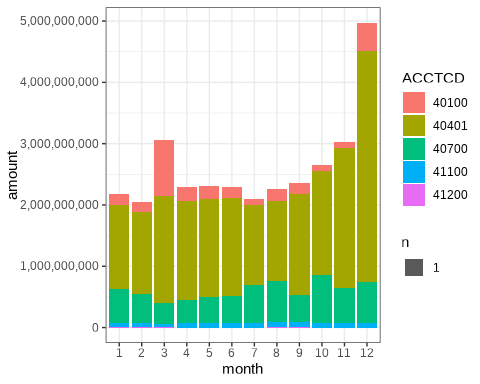
* 교육 수입 월별 추세 분석

acc <- "40700"  
trend(acc)



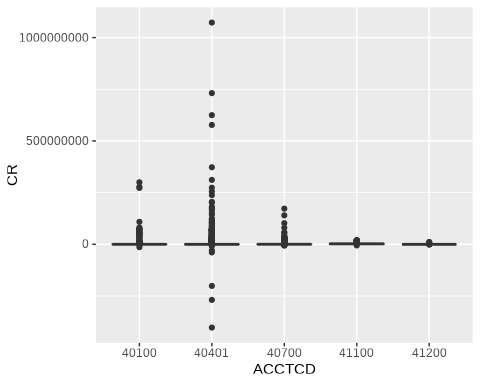
* 총 매출 계정별 월별 추세 분석 - bar plot (stack)

acc <- c("40100", "40401", "40700", "41100", "41200")  
  
je\_tbl |>   
 filter(ACCTCD %in% acc) |>   
 mutate(ACCTCD = as.factor(ACCTCD)) |>   
 mutate(month = as.factor(month(JEDATE))) |>   
 summarise(  
 amount = sum(CR),  
 .by = c(ACCTCD, month)  
 ) |>   
 ggplot(aes(x=month, y=amount, fill=ACCTCD))+  
 geom\_bar(position='stack', stat = "sum")+  
 scale\_y\_continuous(labels = scales::comma)+  
 theme\_bw()



* 총 매출 계정별 box plot - 수치형 변수의 범주별 통계요약

acc <- c("40100", "40401", "40700", "41100", "41200") # 매출 계정 모음  
  
je\_tbl |>   
 filter(ACCTCD %in% acc) |>   
 mutate(ACCTCD = as.factor(ACCTCD)) |>   
 ggplot(aes(x=ACCTCD, y=CR))+  
 geom\_boxplot()



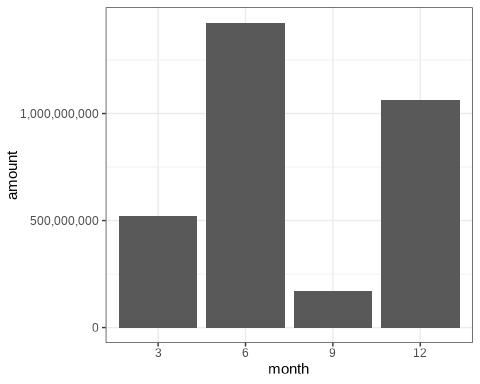
## 제품매출 거래처 갯수 확인

je\_tbl |>  
 filter(ACCT\_NM == '제품매출') |>   
 select(`거래처[코드]`) |>   
 distinct(`거래처[코드]`) |>   
 count()

## # A tibble: 1 × 1  
## n  
## <int>  
## 1 4663

## 원재료비 월별 추세 분석

acc = '50100'  
  
je\_tbl |>   
 filter(ACCTCD==acc) |>   
 mutate(month = as.factor(month(JEDATE))) |>   
 summarise(  
 amount = sum(DR),  
 .by = month  
 ) |> ggplot(aes(x=month, y=amount))+  
 geom\_col()+  
 scale\_y\_continuous(labels = scales::comma)+  
 theme\_bw()



## 매출액 MUS(Monetary Unit Selection) Sampling

* MUS Sampling 함수 정의

## MUS sampling 함수 생성  
mus\_sampling <- \(SR, RC, PL, PM, EA, pop){  
   
 ## Assurance Factor 산정  
   
 assurance\_factor\_raw <- tibble::tribble(  
 ~Significant.Risk, ~Reliance.on.Controls, ~High, ~Moderate, ~Low, ~Analytical.Procedures.Not.Performed,  
 "Yes", "No", 1.1, 1.6, 2.8, 3,  
 "No", "No", 0, 0.5, 1.7, 1.9,  
 "Yes", "Yes", 0, 0.2, 1.4, 1.6,  
 "No", "Yes", 0, 0, 0.3, 0.5  
 )  
   
 assurance\_factor <- assurance\_factor\_raw |>   
 pivot\_longer(  
 cols = c(High, Moderate, Low, Analytical.Procedures.Not.Performed),  
 names\_to = "Planned\_Level", # Planned Level of Assurance from Substantive Analytical Procedures  
 values\_to = "Assurance\_Factor"  
 )  
   
 assurance\_factor <- assurance\_factor |>  
 filter(  
 Significant.Risk == SR,  
 Reliance.on.Controls == RC,  
 Planned\_Level == PL  
 )  
 AF <- assurance\_factor[[1,4]]  
   
 ## Sampling Interval = (Tolerable Misstatement – Expected Misstatement) / Assurance Factor  
 sampling\_interval = (PM - EA) / AF  
   
 ## Consideration of Zero or Negative Amounts  
 pop <- pop |>   
 filter(amount > 0)  
   
 ## High Value  
 high\_value\_items <- pop |>   
 filter(amount >= PM)  
   
 pop\_remain <- pop |>   
 filter(amount < PM)  
   
 ## Expected Sample Size = (Population Subject to Sampling X Assurance Factor) / (Tolerable Misstatement – Expected Misstatement)  
 pop\_amount <- pop\_remain$amount |> sum()  
 sample\_size <- round(pop\_amount \* AF / (PM- EA))  
   
 sampling\_row <- seq(sample\_size)   
 sampling\_n <- seq(sample\_size) \* sampling\_interval  
   
 pop\_remain <- pop\_remain |>   
 mutate(cum = cumsum(amount))  
   
 for (i in seq\_along(sampling\_n)) {  
 sampling\_row[i] <- which(pop\_remain$cum > sampling\_n[i])[1]  
 }  
   
 sampling\_row <- sampling\_row |> unique()  
   
 ## 샘플링 객체 생성  
 sampling\_remain <- pop\_remain |>   
 select(-cum) |>   
 slice(sampling\_row)  
   
 sampling <<- high\_value\_items |>   
 bind\_rows(sampling\_remain)  
}

* Parameters Setting

SR <- "Yes" ## or "No"  
RC <- "Yes" ## or "No"  
PL <- "Analytical.Procedures.Not.Performed" ## or "High", "Low", "Moderate"  
  
## 수행중요성 금액 및 허용오류율(5% 등) 입력  
PM <- 700000000 ## Tolerable misstatement (generally performance materiality)  
EA <- PM \* 0.05 ## Expected misstatement

* 모집단 filter 및 Sampling 수행

## Population  
acc <- c("40100", "40401", "40700", "41100", "41200") # 매출 계정 모음  
  
pop <- je\_tbl |>   
 rename(amount = CR) |> # mus\_sampling 함수 실행을 위하여 금액 열 이름을 amount로 변경.  
 filter(ACCTCD %in% acc) |>   
 mutate(JENO = as.character(JENO))  
  
## 함수 실행  
mus\_sampling(SR, RC, PL, PM, EA, pop) # pop : 샘플링 대상 모집단  
  
## 샘플링 내역 확인  
print(sampling, n = Inf)

## # A tibble: 76 × 22  
## JEDATE `구 분[기표][번호]` JENO LN\_SQ DRCR\_FG ACCTCD ACCT\_NM DR  
## <date> <dbl> <chr> <dbl> <dbl> <dbl> <chr> <dbl>  
## 1 2017-12-14 76 2017121476 2 4 40401 제품매… 0  
## 2 2017-12-14 78 2017121478 2 4 40401 제품매… 0  
## 3 2017-01-10 178 20170110178 2 4 40401 제품매… 0  
## 4 2017-01-20 67 2017012067 2 4 40401 제품매… 0  
## 5 2017-01-25 26 2017012526 2 4 40401 제품매… 0  
## 6 2017-01-31 1111 2017013111… 2 4 40401 제품매… 0  
## 7 2017-01-31 734 20170131734 2 4 40700 교육수… 0  
## 8 2017-02-10 24 2017021024 2 4 40401 제품매… 0  
## 9 2017-02-15 99 2017021599 2 4 40401 제품매… 0  
## 10 2017-02-22 1620 2017022216… 2 4 40401 제품매… 0  
## 11 2017-02-28 240 20170228240 2 4 40401 제품매… 0  
## 12 2017-02-28 765 20170228765 2 4 40401 제품매… 0  
## 13 2017-03-06 1 201703061 3 4 40700 교육수… 0  
## 14 2017-03-13 43 2017031343 2 4 40401 제품매… 0  
## 15 2017-03-13 46 2017031346 2 4 40100 상품매… 0  
## 16 2017-03-20 14 2017032014 2 4 40401 제품매… 0  
## 17 2017-03-24 57 2017032457 2 4 40401 제품매… 0  
## 18 2017-03-31 650 20170331650 2 4 40401 제품매… 0  
## 19 2017-03-31 1062 2017033110… 2 4 40401 제품매… 0  
## 20 2017-03-31 1080 2017033110… 2 4 40700 교육수… 0  
## 21 2017-04-07 57 2017040757 2 4 40401 제품매… 0  
## 22 2017-04-13 15 2017041315 2 4 40401 제품매… 0  
## 23 2017-04-24 1239 2017042412… 2 4 40401 제품매… 0  
## 24 2017-04-27 38 2017042738 2 4 40100 상품매… 0  
## 25 2017-04-30 522 20170430522 2 4 40700 교육수… 0  
## 26 2017-05-02 83 2017050283 2 4 40401 제품매… 0  
## 27 2017-05-15 44 2017051544 2 4 40401 제품매… 0  
## 28 2017-05-23 2254 2017052322… 2 4 40401 제품매… 0  
## 29 2017-05-31 438 20170531438 2 4 40700 교육수… 0  
## 30 2017-05-31 771 20170531771 2 4 40401 제품매… 0  
## 31 2017-05-31 953 20170531953 2 4 40700 교육수… 0  
## 32 2017-06-10 3 201706103 2 4 40401 제품매… 0  
## 33 2017-06-20 906 20170620906 2 4 40401 제품매… 0  
## 34 2017-06-20 3218 2017062032… 2 4 40401 제품매… 0  
## 35 2017-06-30 448 20170630448 2 4 40700 교육수… 0  
## 36 2017-06-30 986 20170630986 2 4 40401 제품매… 0  
## 37 2017-06-30 1035 2017063010… 2 4 40700 교육수… 0  
## 38 2017-06-30 1086 2017063010… 2 4 40401 제품매… 0  
## 39 2017-07-10 86 2017071086 2 4 40401 제품매… 0  
## 40 2017-07-20 1218 2017072012… 2 4 40401 제품매… 0  
## 41 2017-07-31 529 20170731529 2 4 40401 제품매… 0  
## 42 2017-07-31 797 20170731797 2 4 40401 제품매… 0  
## 43 2017-07-31 965 20170731965 2 4 40401 제품매… 0  
## 44 2017-08-04 238 20170804238 3 4 40700 교육수… 0  
## 45 2017-08-20 14 2017082014 2 4 40401 제품매… 0  
## 46 2017-08-24 23 2017082423 2 4 40401 제품매… 0  
## 47 2017-08-31 340 20170831340 2 4 40700 교육수… 0  
## 48 2017-08-31 959 20170831959 2 4 40401 제품매… 0  
## 49 2017-08-31 1116 2017083111… 2 4 40401 제품매… 0  
## 50 2017-09-18 31 2017091831 2 4 40401 제품매… 0  
## 51 2017-09-21 3133 2017092131… 2 4 40401 제품매… 0  
## 52 2017-09-27 39 2017092739 2 4 40100 상품매… 0  
## 53 2017-09-30 699 20170930699 2 4 40700 교육수… 0  
## 54 2017-09-30 907 20170930907 2 4 40401 제품매… 0  
## 55 2017-10-10 293 20171010293 2 4 40401 제품매… 0  
## 56 2017-10-18 20 2017101820 2 4 40401 제품매… 0  
## 57 2017-10-25 51 2017102551 2 4 40700 교육수… 0  
## 58 2017-10-31 691 20171031691 2 4 40401 제품매… 0  
## 59 2017-10-31 766 20171031766 2 4 40700 교육수… 0  
## 60 2017-10-31 980 20171031980 2 4 40401 제품매… 0  
## 61 2017-10-31 1045 2017103110… 2 4 40700 교육수… 0  
## 62 2017-11-10 133 20171110133 2 4 40401 제품매… 0  
## 63 2017-11-20 1876 2017112018… 2 4 40401 제품매… 0  
## 64 2017-11-30 681 20171130681 2 4 40700 교육수… 0  
## 65 2017-11-30 1067 2017113010… 2 4 40700 교육수… 0  
## 66 2017-11-30 1156 2017113011… 2 4 40401 제품매… 0  
## 67 2017-11-30 1170 2017113011… 2 4 40401 제품매… 0  
## 68 2017-11-30 1171 2017113011… 2 4 40401 제품매… 0  
## 69 2017-12-10 8 201712108 2 4 40401 제품매… 0  
## 70 2017-12-18 309 20171218309 2 4 40401 제품매… 0  
## 71 2017-12-22 28 2017122228 2 4 40401 제품매… 0  
## 72 2017-12-26 611 20171226611 2 4 40100 상품매… 0  
## 73 2017-12-28 90 2017122890 2 4 40100 상품매… 0  
## 74 2017-12-31 832 20171231832 2 4 40700 교육수… 0  
## 75 2017-12-31 1026 2017123110… 2 4 40700 교육수… 0  
## 76 2017-12-31 1269 2017123112… 2 4 40401 제품매… 0  
## # ℹ 14 more variables: amount <dbl>, `거래처[코드]` <chr>,  
## # `구 분[결의][년/월/일]` <dbl>, `구 분[결의][번호]` <dbl>,  
## # `결의부서[코드]` <dbl>, `결의사원[코드]` <chr>, `PJT[코드]` <dbl>,  
## # `유형번호[번호]` <dbl>, `유형번호[유형명]` <chr>, `증빙구분[구분명]` <chr>,  
## # INSERT\_ID <chr>, 입력일자 <chr>, MODIFY\_ID <chr>, MODIFY\_DT <chr>

## 추출된 샘플 엑셀 파일 생성  
sampling |> write\_xlsx("sample.xlsx")

## 분개장 전표에서 매출 계정별 원장 추출 - 엑셀로 저장

acc <- c("40100", "40401", "40700", "41100", "41200")  
  
gl <- je\_tbl %>%   
 filter(ACCTCD %in% acc) %>%  
 split(.$ACCT\_NM)  
  
write\_xlsx(gl, path = "gl.xlsx")