IDL에서 기초통계를 해 봅시다

Intro

IDL에서는 각종 수치계산 기능들이 제공되는데 그 중 통계를 위한 기능함수들이 있습니다. IDL 도움말의 Contents 탭에서 Routines (by topic) → Mathematics 항목으로 가서 Statistical Tools라는 섹션을 보면 관련 내장함수들이 소개되어 있습니다. 그 중 기초 통계량을 계산하는 관련 함수들을 예제와 함께 살펴보기로 하겠습니다.

MOMENT 함수

MOMENT 함수를 사용하면 여러 종류의 기초통계량들을 한꺼번에 구할 수 있습니다. 디폴트로 사용하면 4종, 그리고 추가 키워드들을 사용하면 6종의 통계값들을 계산할 수 있습니다. 다음과 같이 11개의 값들로 구성된 배열 data에 대하여 MOMENT 함수를 디폴트 설정만으로 사용해봅시다.

```
data = [13, 8, 11, 11, 7, 8, 15, 10, 8, 9, 6]
result = MOMENT(data)
```

이와 같이 디폴트로 사용하면 4개의 값들로 구성된 배열을 되돌려주는데 이 배열은 [평균(mean), 분산(variance), 왜도(skewness), 첨도(kurtosis)]의 형태로 구성됩니다.

```
print, result
    9.63636     7.25455    0.547464   -0.914356
```

여기서 SDEV, MDEV 키워드를 추가적으로 사용하면 표준편차(standard deviation), 평균절대편차(mean absolute deviation) 값들도 얻을 수 있습니다.

```
result = MOMENT(data, SDEV=sd, MDEV=mad)
PRINT, sd, mad
   2.69343  2.14876
```

MOMENT 함수에서 통계량들을 계산하는데 사용된 공식은 IDL 도움말의 MOMENT 함수에 관한 내용에서 확인할 수 있습니다.

개별 통계량별 함수

MOMENT 함수를 사용하여 여러 종의 통계량들을 한꺼번에 얻는 대신, 각 통계량별 전용함수를 사용하여 개별적으로 얻는 것도 가능합니다. 주요 함수들은 다음과 같습니다.

- MEAN : 평균값 (산술평균)
- MEDIAN : 중간값
- VARIANCE : 분산 (표본)
- STDDEV : 표준편차 (표본)
- SKEWNESS : 왜도
- KURTOSIS : 첨도
- MEANABSDEV : 평균 절대 편차
- TOTAL : 총합
- MIN : 최소값
- MAX : 최대값
- N_ELEMENTS : 갯수

따라서 위의 함수들을 사용하여 결과를 얻어 보면 다음과 같습니다.

```
PRINT, MEAN(data)
   9.63636
PRINT, MEDIAN(data)
   9.00000
PRINT, VARIANCE(data)
   7.25455
PRINT, STDDEV(data)
   2.69343
PRINT, SKEWNESS(data)
   0.547464
PRINT, KURTOSIS(data)
   -0.914356
PRINT, MEANABSDEV(data)
   2.14876
PRINT, TOTAL(data)
   106.000
PRINT, MIN(data)
   6
PRINT, MAX(data)
   15
PRINT, N_ELEMENTS(data)
   11
```

소팅(Sorting)

배열 내 값들을 올림차순 또는 내림차순으로 정렬하는 작업으로서 IDL에서는 SORT 함수가 그 기능을 제공합니다. 사용 예제는 다음과 같습니다.

```
st = SORT(data)
data_sorted = data[st]
PRINT, data_sorted
   6 7 8 8 8 9 10 11 11 13 15
```

이와 같이 SORT 함수로 얻은 결과로 원래 데이터 배열을 인덱싱하면 오름차순으로 정렬된 결과를 얻게 됩니다. 만약 내림차순으로 정렬된 결과를 얻고자 한다면 다음과 같이 REVERSE 함수를 사용하면 됩니다.

```
data_sorted = REVERSE(data[st])
PRINT, data_sorted
  15  13  11  11  10  9  8  8  8  7  6
```

IMSL_SIMPLESTAT 함수

이 함수는 IDL의 add-on 모듈인 IMSL 라이브러리(IDL Advanced Math and Stats)에서 지원되며, 앞서 소개된 MOMENT 함수와 유사하지만 더 다양한 통계값들을 산출해줍니다. IMSL 라이선스가 추가로 설치되어 있는 IDL에서는 다음과 같이 사용할 수 있습니다.

```
result = IMSL_SIMPLESTAT(data)
PRINT, result
```

이렇게 하여 얻어진 결과 배열인 result는 총 14종의 각종 통계값들로 구성됩니다. IDL 도움말에 의하면 이 14종의 통계값들은 다음 표와 같습니다.

i	Statistic Returned in Element (*i*, *)
0	Mean
1	Variance
2	Standard deviation
3	Coefficient of skewness
4	Coefficient of excess (kurtosis)
5	Minimum value
6	Maximum value
7	Range
8	Coefficient of variation (when defined). If the coefficient of variation is not defined, zero is returned.
9	Number of observations (the counts)
10	Lower confidence limit for the mean (assuming normality). The default is a 95-percent confidence interval.
11	Upper confidence limit for the mean (assuming normality)
12	Lower confidence limit for the variance (assuming normality). The default is a 95-percent confidence interval.
13	Upper confidence limit for the variance (assuming normality)

그리고 실제로 출력된 14개의 값들은 다음과 같습니다.

```
    9.63636     7.25454     2.69343     0.631604
   -0.476370    6.00000     15.0000     9.00000
    0.279506    11.0000     7.82690     11.4458
    3.54171     22.3425
```

RANDOMU와 RANDOMN

난수(RANDOM)의 생성은 모의데이터를 만들기 위해서 흔히 사용하는 방법입니다. IDL에서는 균일분포(Uniform Distribution) 기반의 난수들을 발생시키는 RANDOMU 함수와 정규분포(Normal Distribution) 기반의 난수들을 발생시키는 RANDOMN 함수 두 종류가 제공됩니다. 이 함수들의 세부적인 사용법은 도움말에 잘 나와있습니다. RANDOM 함수에서 Seed 값은 고정값(예: -1)을 주어 늘 같은 난수가 나오게 할 수도 있고, 변수를 이용하여 매번 새로운 난수가 나오게 할 수도 있습니다. (IDL Advanced Math and Stats 라이선스가 있는 경우 IMSL_RANDOM을 사용할 수도 있습니다.)

- RANDOMU(seed, 10000) : 0.0~1.0 범위의 균일 분포 난수 10000개를 생성합니다. 0.0~2.0 범위의 난수를 만들려면 곱하기 2를 하면 됩니다. -0.5~0.5 범위의 난수를 만들려면 0.5를 빼면 됩니다.
  이를 이용하여, -1.0~1.0 범위의 난수 10000개를 만드는 방법은 다음과 같습니다.
  uran=RANDOMU(seed, 10000) * 2.0 - 1.0
- RANDOMN(seed, 10000) : 평균 0.0, 표준편차 1.0인 정규분포 난수 10000개를 발생합니다. 평균이 x인 정규분포를 만들려면 x를 더하면 되고, 표준편차가 s 인 정규분포를 만들려면 s를 곱하면 됩니다.
  이를 이용하여, 평균이 3, 표준편차가 2인 정규분포 난수 10000개를 만드는 방법은 다음과 같습니다.
  nran=RANDOMN(seed, 10000) * 2.0 + 3.0

```
uran=RANDOMU(seed, 10000)*2.0-1.0
PRINT, MEAN(uran), MIN(uran), MAX(uran)
    0.00557242      -0.999892       0.999871
```

위의 코드로 생성한 균일분포 난수들은 개수가 많아질수록, 평균은 0에, 최대 최소는 1.0, -1.0에 근접할 것입니다. 그리고 아래 코드로 생성한 정규분포 난수들은 개수가 많아질수록, 평균은 3.0에, 표준편차는 2.0에 근접할 것입니다.

```
nran=RANDOMN(seed, 10000) * 2.0 + 3.0
PRINT, MEAN(nran), STDDEV(nran)
    3.00635      2.01185
```