

# parallel processing in sas

November 5, 2019

## 1 parallel processing in sas

torsten dahlén

### 2 outline

- background, terminology
- in sas, in general
- method: “poor mans” parallel processing at kep in sas
- then i think jing has some quiz to increase “knowledge retention”

#### 2.1 recap ‘parallel processing’ from helga

- to perform independent tasks in parallel rather than consecutively

dataset A --> t\_1 --> t\_2 --> ... -> t\_n --> Output

- in parallel

dataset A --> t\_1 --> OutputB1

dataset A --> t\_2 --> OutputB2

dataset A --> t\_n --> OutputBn

----> Merge OutputB:

#### 2.2 basics: threads vs cpus vs parallel processing

- threads = number of concurrent processes. single cpu can handle multiple tasks at the same time but -> depends on clock-speed -> increased heat production -> inefficient processing.
- cpu cores = threading on different cores -> less energy expenditure -> more efficient processing
- parallel processing = break down tasks to smaller units to be processed in parallel.

## 2.3 tasks are either

- i/o bound = the read/write speeds are limiting the task, solution = get multiple ssd's that sas can "thread" data to
- cpu/process bound = the cpu utilization is limiting the task, solution = threading

## 2.4 gains: amdahl's law

- the decrease in computation time is limited by tasks that cannot be parallelized
- thus, slightly decreasing returns with increased number of cpus because of tasks that cannot be parallelized

## 2.5 do i need to parallelize my code?

- `options fullstimer;` = get full log information on time and memory consumption to identify possible shortcomings in your resources.
- run task manager to see resource usage.
- ...but simply do i perform time-consuming tasks on my datasets? are they independent?

## 2.6 how many cpus do i have according to sas?

```
%put (&sysncpu);  
(4)
```

## 2.7 parallel processing in sas

- sas base has a few built-in proc's that offer the use of multiple threads.

```
proc means  
proc report  
proc sort  
proc summary  
proc tabulate  
proc sql
```

- options: CPUCOUNT= specifies how many CPUs can be used, can be set to numeric value or ACTUAL  
THREAD|NOTHEADS = controls whether to use threads.

If the THREADS system option is set to NOTHEADS, the CPUCOUNT= option has no effect.

## 2.8 parallel processing in sas ii

- in sas stat the following procedures have abilities to use multiple threads

```
proc adaptivereg
proc fmm
proc glm
proc glmselect
proc loess
proc mixed
proc quantlife
proc quantreg
proc quantselect
proc rubustreg
```

## 2.9 parallel processing in sas iii : example

```
options threads cpucount=4;
```

```
proc sql;
create table blaha as select
a.* from big_data_a a inner join big_data_b b
on a.id=b.id
where a.id in (select distinct id from super_large_data where huge_list_of_obs between 3222 and
order by a.id;
quit;
```

## 2.10 how can i do parallel processing in sas at kep?

- “poor mans” version
- local sas/kep-sas-servers (*no sas connect with remote submit*)
- we can use systask

## 2.11 what is systask ?

- = launch and handle shell tasks from within sas (r has `shell()` function)
- **very simple: launch concurrent sas sessions from within sas and tell sas when they are completed**

*can be used for pretty much anything: use `systask` to launch spotify, use instead of `proc iml` (needs `kep_admin` password to configure) to run r from sas*

- `systask` works in linux (also x command), small differences between versions of sas, read man-pages

## 2.12 framework for parallel processing

```
libname temp 'path';
```

- move dataset that are used to your temp-path
- generate .sas to be processed in parallel and where the same libname is called to access same path
- run a systask eg

```
systask command "sas H:\temp\a1.sas" nowait taskname=a1;  
systask command "sas H:\temp\a2.sas" nowait taskname=a2;  
waitfor _all_ a1 a2;
```

- do whatever with resulting datasets
- log files will be created if you specify -log and location, eg

```
systask command "sas H:\temp\a1.sas -log H:\logs\log.log" nowait taskname=a1;
```

## 2.13 live example using the code challenge data and task

- read dataset from .csv
- very simplistic case-control study, estimate or for exposure to pneumonia and/or mononucleosis on the risk of ms using logistic regression.
- what parts of this could be parallelized?

import csv? create studybase (casecontrols and diagnosis information) ? categorize into age groups? sum groups? dichotomize exposures? running the logistic regression? calculate age-specific means?