Github repository with your code (complete URL):

	Optimization+platform dependent options for compilation and execution times for `my_dgesv`	
	icc (18.0.1)	**gcc (8.1.0)**
No opt.	Options: -O0 Exec time (small): 394.40s Exec time (medium): 3754s Exec time (large): 35838s	Options: -O0 Exec time (small): 367.47s Exec time (medium): 3720s Exec time (large): 35406s
Opt level 1	Options: -O1 Exec time (small): 163.78s Exec time (medium): 3443s Exec time (large): 34107s	Options: -O1 Exec time (small): 162.29s Exec time (medium): 3423s Exec time (large): file problem
Opt level 2 + specific arch	Options: -O2 -march=native Exec time (small): 109.06s Exec time (medium): 2778s Exec time (large):	Options: -O2 -march=native Exec time (small): 165.89s Exec time (medium): 3479s Exec time (large):
Opt level 3 + specific arch	Options: -O3 -march=native Exec time (small): 108.20s Exec time (medium): 2730s Exec time (large):	Options: -O3 -march=native Exec time (small): 167.08s Exec time (medium): 3443s Exec time (large):
Opt level fast + specific arch	Options: -Ofast -march=native Exec time (small): 111.35s Exec time (medium): 2722s Exec time (large): 15348s	Options: -Ofast -march=native Exec time (small): 165.44s Exec time (medium): 3530s Exec time (large): 20865
Opt level fast + specific arch + interproc opt/anal [ipo (icc) / -fipa-pta (gcc)]	Options: -Ofast -march=native -ipo Exec time (small): 120.17s Exec time (medium): 2705s Exec time (large): 28055s	Options: Ofast -march=native -fipa-pta Exec time (small): 168.56s Exec time (medium): 3454s Exec time (large): 20144s
All previous opts + pgo	Options: -Ofast -march=native -ipo -prof-gen Exec time (small): 202.55s Exec time (medium): 3446s Exec time (large): 35140s	Options: Ofast -march=native -fipa-pta -fprofile-generate Exec time (small): 157.94s Exec time (medium): 3495s Exec time (large):
Others: autovectorizing? Autoparallelism?	Options: Exec time (small): Exec time (medium): Exec time (large):	Options: Exec time (small): Exec time (medium): Exec time (large):

Matrix A size for executions, according to `Makefile`:

- small size: 2048 x 2048 (execute as ./dgesv 2048)
- medium size: 4096 x 4096 (execute as ./dgesv 4096)
- large size: 8192 x 8192 (execute as ./dgesv 8192)

Execute at least 3 times per combination, taking the middle value (median). The relevant time is the execution time for your implementation of `my_dgesv` (i.e. your solver!).

Description and relevant information extracted from the results:

Small test:

We can see that without any option, the execution time with a gcc or icc compilation is similar, the execution after a gcc compilation is just a bit faster.
However, when we add options for compilation, the execution time after an icc compilation is better than with a gcc compilation. The time is better for all different options except for the level 1 which gets the same time as with a gcc compilation. Furthermore, when we add the pgo, for compilation with icc, the execution time increases compared to without options.
To conclude, the compilation with icc is globally more efficient than the compilation with gcc for the small test.

Medium test:

For the medium test, without option the both compilation have almost the same execution time but when we use an option, the execution time for the icc compilation has a better execution time than the gcc compilation. We can notice that it is not interesting to add options for the gcc compilation because the execution time doesn't change much.
To conclude, for this test, it is not very useful to compile the code with any option for the gcc compilation but we should use any option (doesn't really matter which one, except level 1) for the icc compilation.

Large test:

For the large test, I noticed that that both compilation are similar and it is useful to use option during the compilation. However, I got some problems of compilation and because of the high execution time, I don't have enough time to compile and execute another time the files which didn't work.
I had different problem: "file problem", "dgesv.c does not exist", "text file busy".