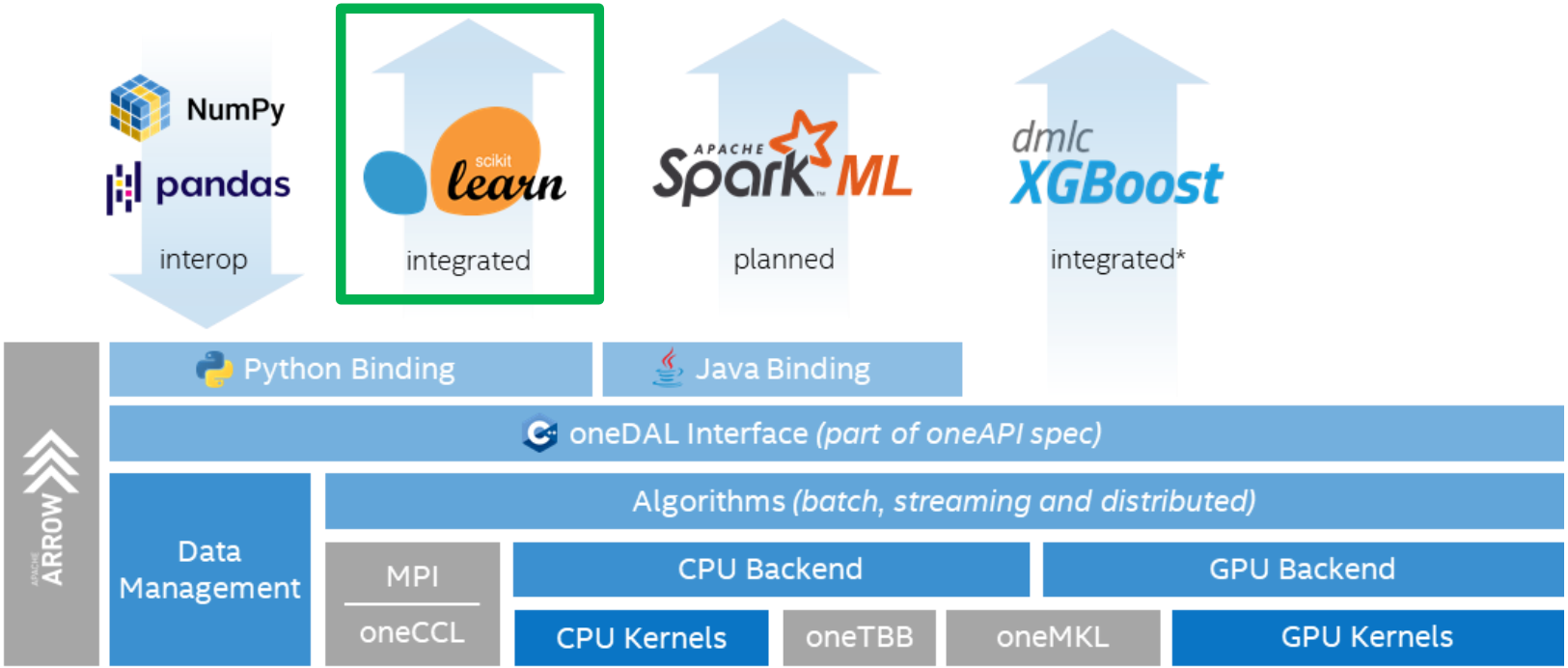

intel
Software
SPEEDUP CLASSIC MACHINE LEARNING
AND DATA ANALYTICS WITH INTEL

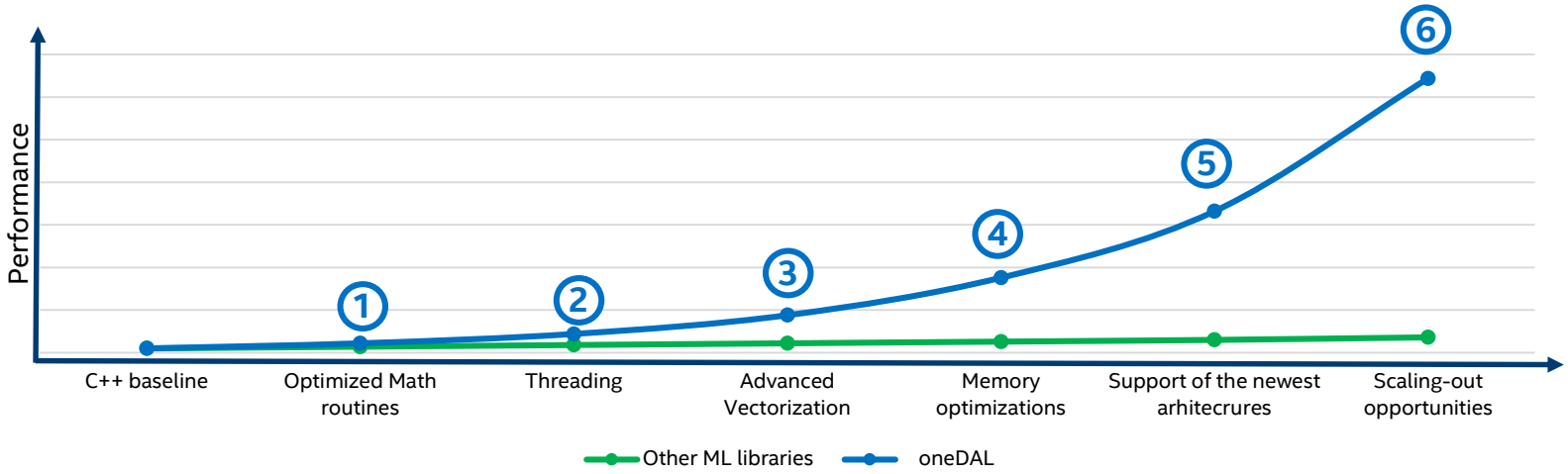
oneAPI Data Analytics Library (oneDAL)

Optimized building blocks for all stages of data analytics on Intel Architecture



GitHub: https://github.com/oneapi-src/oneDAL

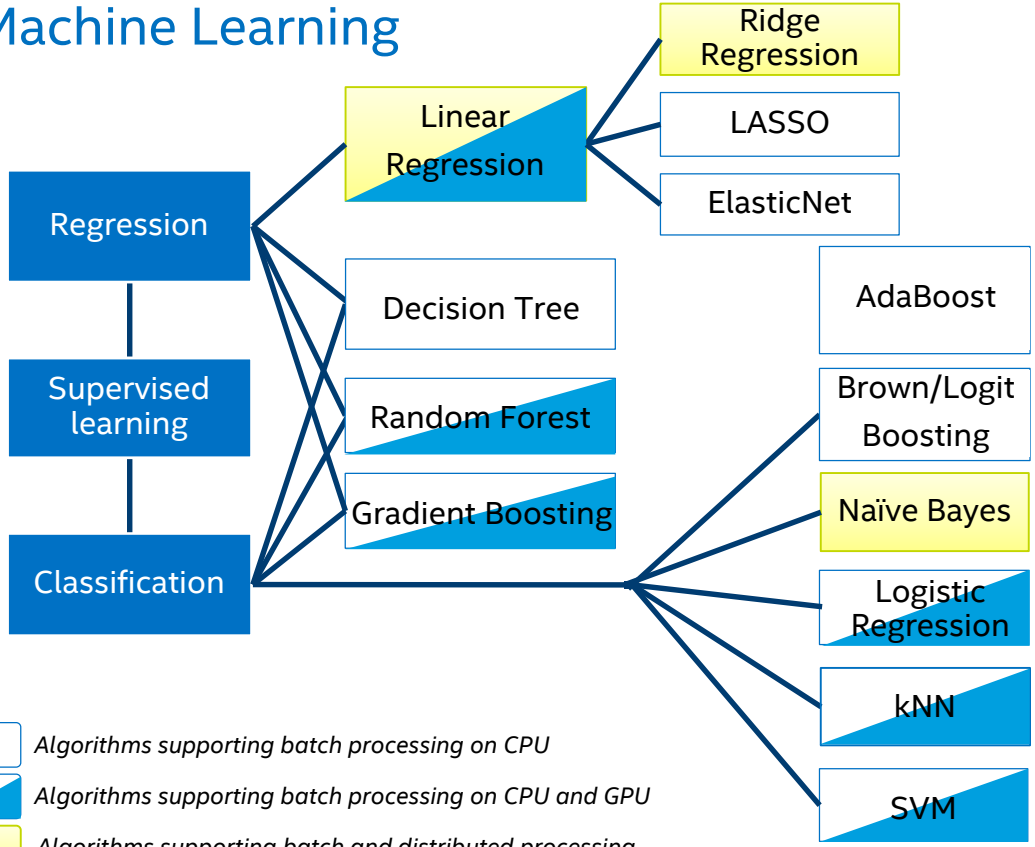
What makes oneDAL faster?

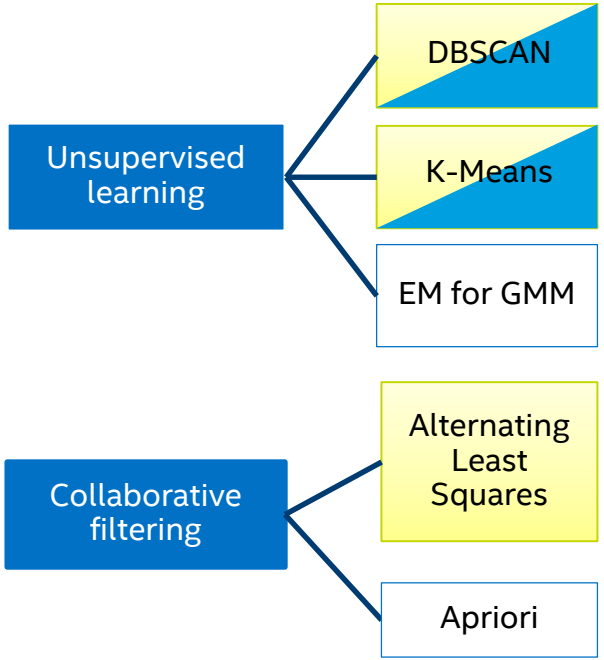


1. The best performance on Intel Architectures with oneMKL (Intel® MKL) vs. less performance OS BLAS/LAPACK libs
2. oneDAL targets to many-core systems to achieve the best scalability on Intel® Xeon, other libs mostly target to client versions with small amount of cores
3. oneDAL uses the latest available vector-instructions on each architecture, enables them by compiler options, intrinsics. Usually other ML libs build application without vector-instructions support or sse4.2 only.
4. oneDAL uses the most efficient memory optimization practices: minimally access memory, cache access optimizations, SW memory prefetching. Usually Other ML libs don't make low-level optimizations.
5. oneDAL enables new instruction sets and other HW features even before official HW lunch. Usually other ML libs do this with long delay.
6. oneDAL provides distributed algorithms which scale on many nodes

oneDAL Algorithms

Machine Learning



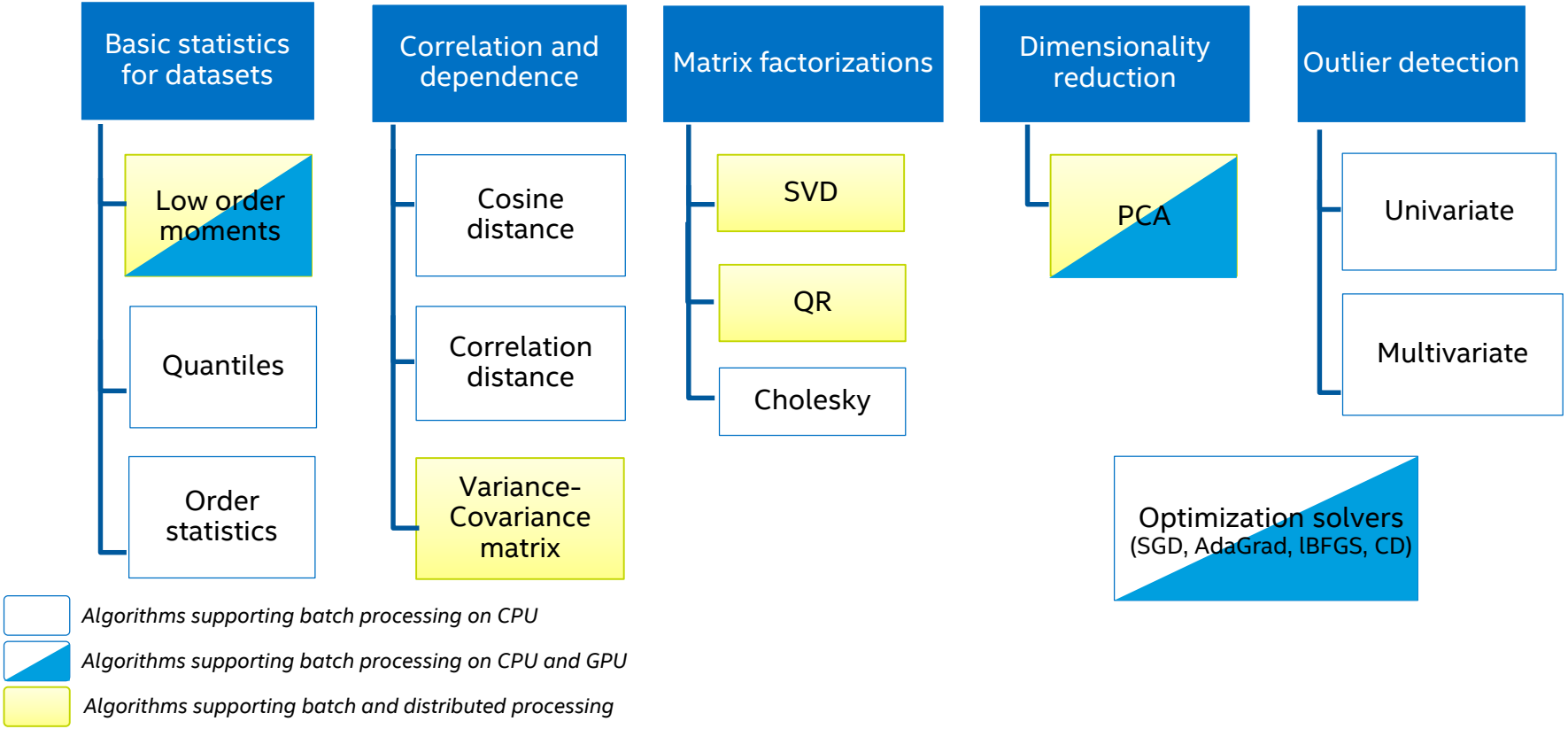


Algorithms supporting batch processing on CPU

Algorithms supporting batch processing on CPU and GPU

Algorithms supporting batch and distributed processing

oneDAL Algorithms
Data Transformation and Analysis



SCIKIT-LEARN ACCELERATION

Intel Distribution for Python (IDP) Scikit-learn

Common Scikit-learn

```
from sklearn.svm import SVC

X, Y = get_dataset()


clf = SVC().fit(X, y)
res = clf.predict(X)
```

Scikit-learn mainline

Scikit-learn with Intel CPU opts

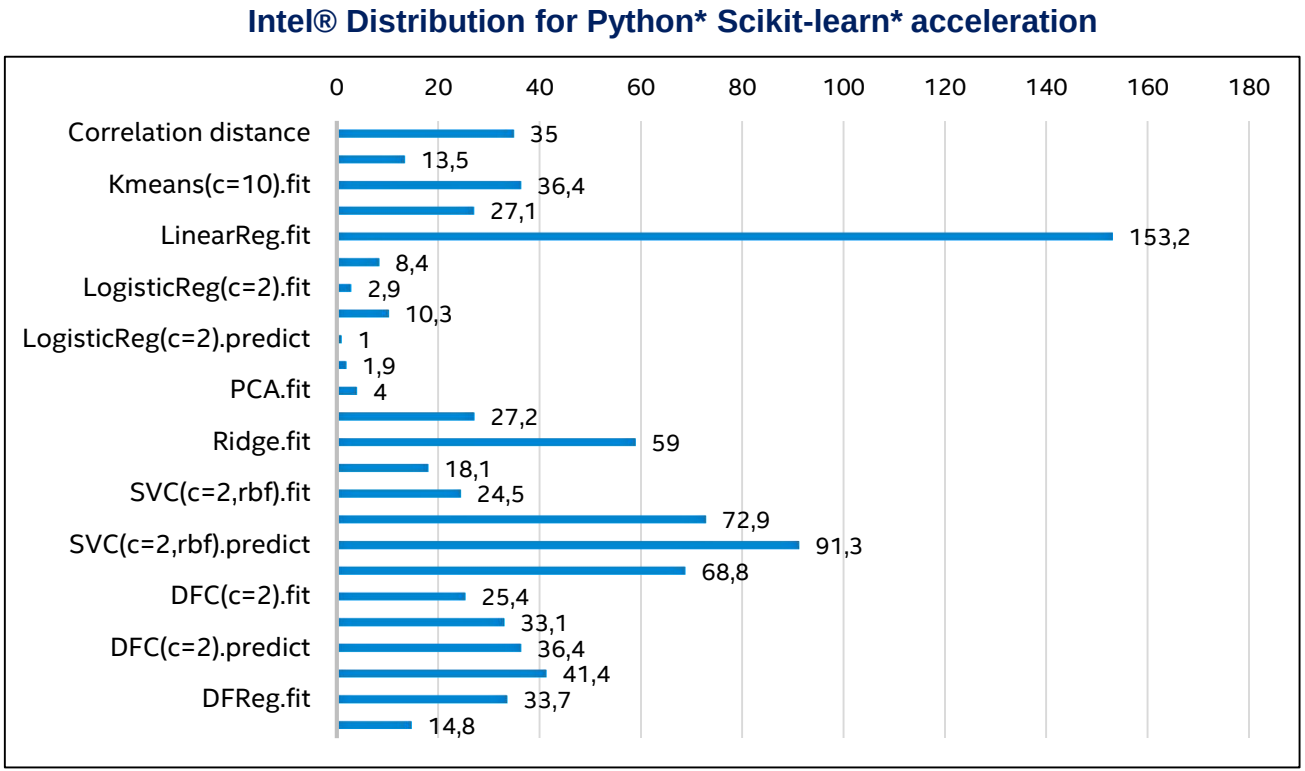
```
import daal4py as d4p
d4p.patch_sklearn()
from sklearn.svm import SVC

X, Y = get_dataset()


clf = SVC().fit(X, y)
res = clf.predict(X)
```

Available through Intel conda
(conda install daal4py -c intel)

Same Code,
Same Behavior

- Scikit-learn, not scikit-learn-*like*
- Scikit-learn conformance (mathematical equivalence) defined by Scikit-learn Consortium, continuously vetted by public CI

Intel optimized Scikit-Learn



Same Code,
Same Behavior

PASSED

- Scikit-learn, not scikit-learn-*like*
- Scikit-learn conformance (mathematical equivalence) defined by Scikit-learn Consortium, continuously vetted by public CI

HW: Intel Xeon Platinum 8276L CPU @ 2.20GHz, 2 sockets, 28 cores per socket;

Details: https://medium.com/intel-analytics-software/accelerate-your-scikit-learn-applications-a06cacf44912

Available algorithms

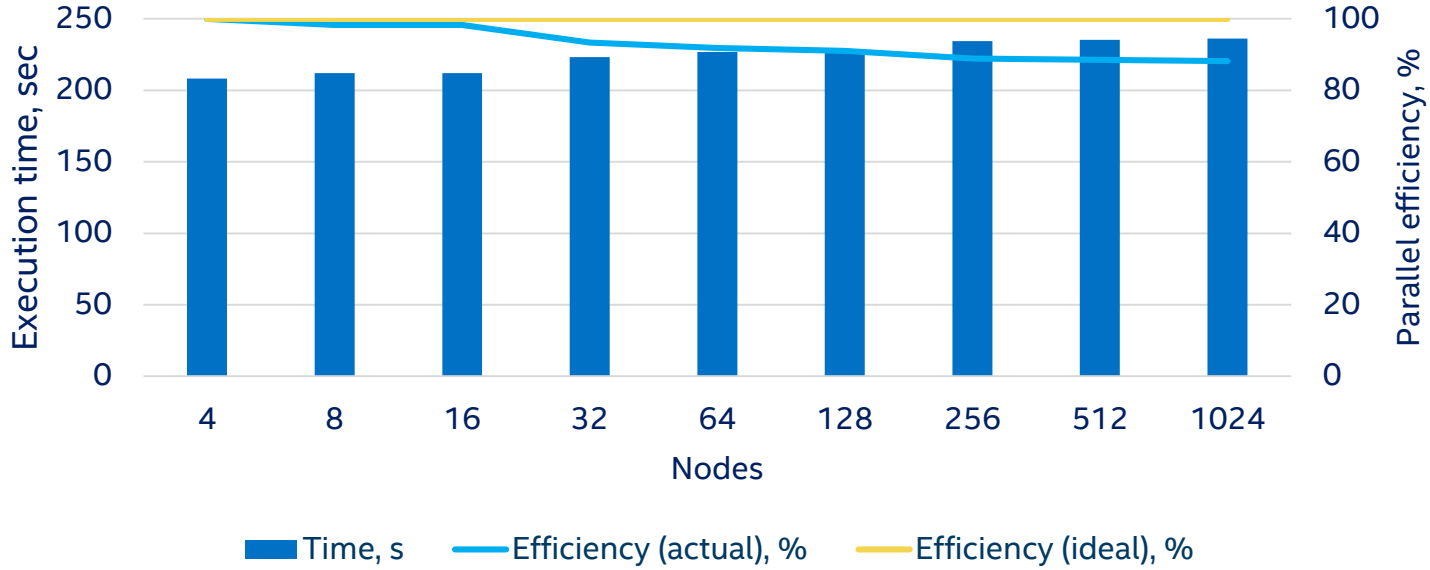
Accelerated IDP Scikit-learn algorithms:

- Linear/Ridge Regression
- Logistic Regression
- ElasticNet/LASSO
- PCA
- K-means
- DBSCAN
- SVC
- train_test_split(), assume_all_finite()
- Random Forest Regression/Classification - DAAL 2020.3
- kNN (kd-tree and brute force) - DAAL 2020.3

SCALE OUT ACCELERATION

Intel® DAAL K-means fit, week scaling results

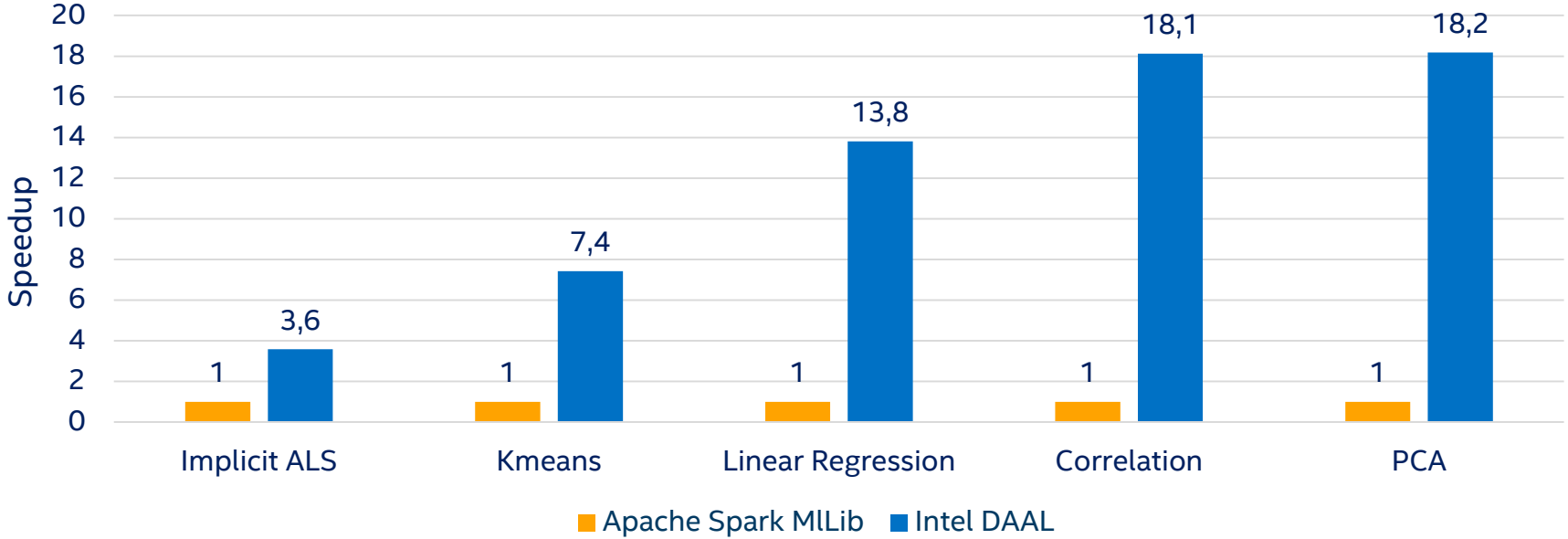
(87.44GB/node, 84 features, 8 clusters, 100 iterations, float32)



Source: https://arxiv.org/abs/1909.11822

Intel technologies' features and benefits depend on system configuration and may require enabled hardware, software or service activation. Learn more at intel.com, or from the OEM or retailer.
Performance results are based on testing as of **09/25/2019** and may not reflect all publicly available security updates. See configuration disclosure for details. No product can be absolutely secure.
Software and workloads used in performance tests may have been optimized for performance only on Intel microprocessors. Performance tests, such as SYSmark and MobileMark, are measured using specific computer systems, components, software, operations and functions. Any change to any of those factors may cause the results to vary. You should consult other information and performance tests to assist you in fully evaluating your contemplated purchases, including the performance of that product when combined with other products. For more complete information visit www.intel.com/benchmarks.
Configuration: Testing by Intel as of **09/25/2019**. 7 x m5.2xlarge AWS instances, Intel® Data Analytics Acceleration Library 2019.3 (Intel® DAAL); Intel Xeon Processor E5-2698 v3 @ 2.3GHz, 2 sockets, 16 cores per socket, MPI4Py (3.0.0), Intel® Distribution Of Python (IDP) 3.6.8, float, Source: https://arxiv.org/abs/1909.11822
Intel's compilers may or may not optimize to the same degree for non-Intel microprocessors for optimizations that are not unique to Intel microprocessors. These optimizations include SSE2, SSE3, and SSSE3 instruction sets and other optimizations. Intel does not guarantee the availability, functionality, or effectiveness of any optimization on microprocessors not manufactured by Intel. Microprocessor-dependent optimizations in this product are intended for use with Intel microprocessors. Certain optimizations not specific to Intel microarchitecture are reserved for Intel microprocessors. Please refer to the applicable product User and Reference Guides for more information regarding the specific instruction sets covered by this notice. Notice revision #20110804

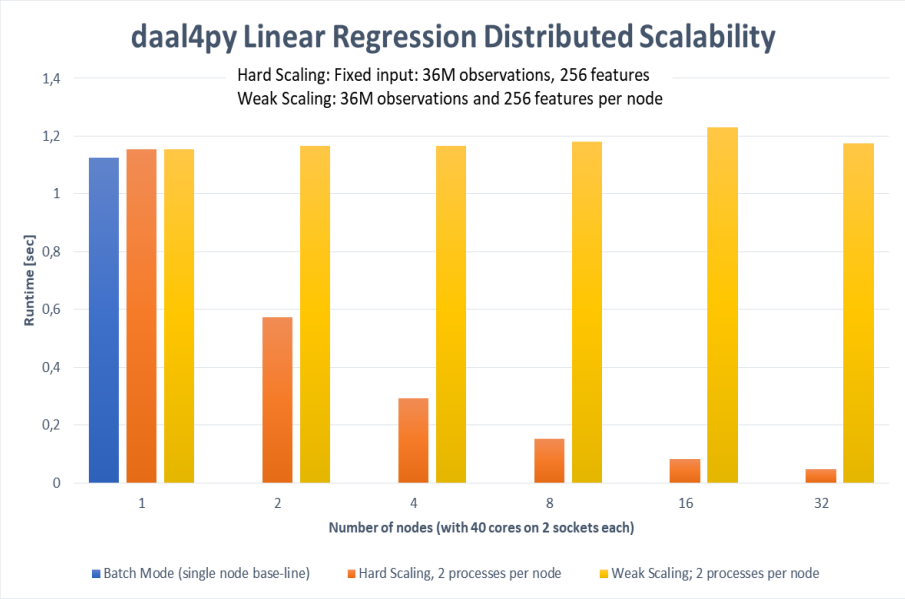
Intel® DAAL 2020 vs Apache Spark* MlLib performance (Higher is better)

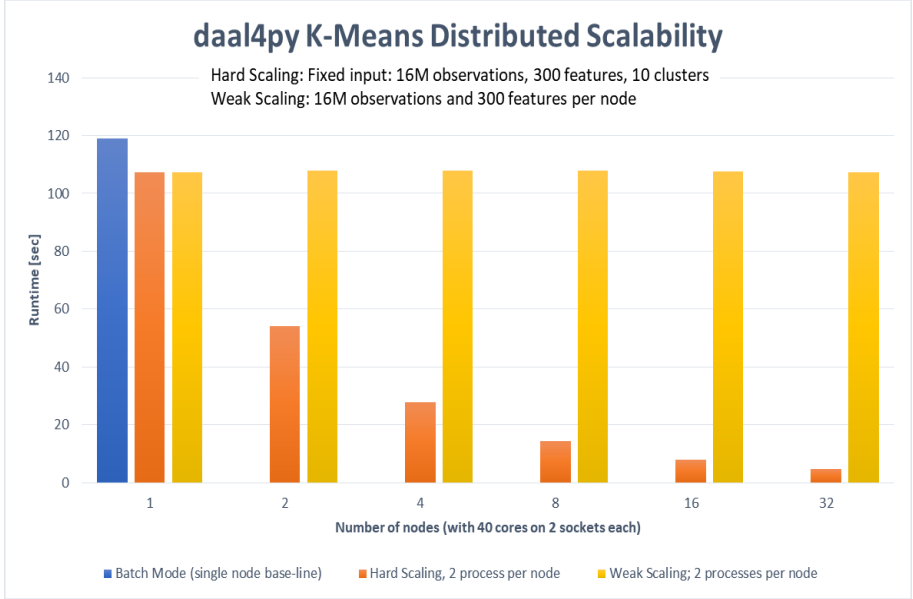


Intel technologies' features and benefits depend on system configuration and may require enabled hardware, software or service activation. Learn more at intel.com, or from the OEM or retailer.
Performance results are based on testing as of **11/11/2019** and may not reflect all publicly available security updates. See configuration disclosure for details. No product can be absolutely secure.
Software and workloads used in performance tests may have been optimized for performance only on Intel microprocessors. Performance tests, such as SYSmark and MobileMark, are measured using specific computer systems, components, software, operations and functions. Any change to any of those factors may cause the results to vary. You should consult other information and performance tests to assist you in fully evaluating your contemplated purchases, including the performance of that product when combined with other products. For more complete information visit www.intel.com/benchmarks.
Configuration: Testing by Intel as of **11/11/2019**. 7 x m5.2xlarge AWS instances, Intel® Data Analytics Acceleration Library 2020 (Intel® DAAL); Correlation (# samples = 10M, # features = 1000, (Intel® DAAL=35.2s, MLLib=638.2s)), PCA (# samples = 10M, # features = 1000 (Intel® DAAL=35.2s, MLLib=639.8s)), implicit ALS (# users = 1M, # items = 1M, # factors = 100, # Iterations = 1 (Intel® DAAL=37.6s, MLLib=134.9s)), Linear Regression (# samples = 100M, # features = 50 (Intel® DAAL=16.3s, MLLib=224.5s)), k-means (# samples = 100M, # features = 50, # clusters = 10, # Iterations = 100 (Intel® DAAL=211s, MLLib=1567.3s))
Intel's compilers may or may not optimize to the same degree for non-Intel microprocessors for optimizations that are not unique to Intel microprocessors. These optimizations include SSE2, SSE3, and SSSE3 instruction sets and other optimizations. Intel does not guarantee the availability, functionality, or effectiveness of any optimization on microprocessors not manufactured by Intel. Microprocessor-dependent optimizations in this product are intended for use with Intel microprocessors. Certain optimizations not specific to Intel microarchitecture are reserved for Intel microprocessors. Please refer to the applicable product User and Reference Guides for more information regarding the specific instruction sets covered by this notice. Notice revision #20110804

Working in distributed environment

Hardware	Intel(R) Xeon(R) Gold 6148 CPU @ 2.40GHz, EIST/Turbo on
	2 sockets, 20 Cores per socket
	192 GB RAM
	16 nodes connected with Infiniband
Operating System	Oracle Linux Server release 7.4
Data Type	double





On a 32-node cluster (1280 cores) daal4py computed linear regression of 2.15 TB of data in 1.18 seconds and 68.66 GB of data in less than 48 milliseconds.

On a 32-node cluster (1280 cores) daal4py computed K-Means (10 clusters) of 1.12 TB of data in 107.4 seconds and 35.76 GB of data in 4.8 seconds.

INTEL DAAL USAGE EXAMPLES

Legal Disclaimer & Optimization Notice

INFORMATION IN THIS DOCUMENT IS PROVIDED “AS IS”. NO LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE, TO ANY INTELLECTUAL PROPERTY RIGHTS IS GRANTED BY THIS DOCUMENT. INTEL ASSUMES NO LIABILITY WHATSOEVER AND INTEL DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY, RELATING TO THIS INFORMATION INCLUDING LIABILITY OR WARRANTIES RELATING TO FITNESS FOR A PARTICULAR PURPOSE, MERCHANTABILITY, OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT.

Software and workloads used in performance tests may have been optimized for performance only on Intel microprocessors. Performance tests, such as SYSmark and MobileMark, are measured using specific computer systems, components, software, operations and functions. Any change to any of those factors may cause the results to vary. You should consult other information and performance tests to assist you in fully evaluating your contemplated purchases, including the performance of that product when combined with other products.

Optimization Notice

Intel’s compilers may or may not optimize to the same degree for non-Intel microprocessors for optimizations that are not unique to Intel microprocessors. These optimizations include SSE2, SSE3, and SSSE3 instruction sets and other optimizations. Intel does not guarantee the availability, functionality, or effectiveness of any optimization on microprocessors not manufactured by Intel. Microprocessor-dependent optimizations in this product are intended for use with Intel microprocessors. Certain optimizations not specific to Intel microarchitecture are reserved for Intel microprocessors. Please refer to the applicable product User and Reference Guides for more information regarding the specific instruction sets covered by this notice.

Notice revision #20110804