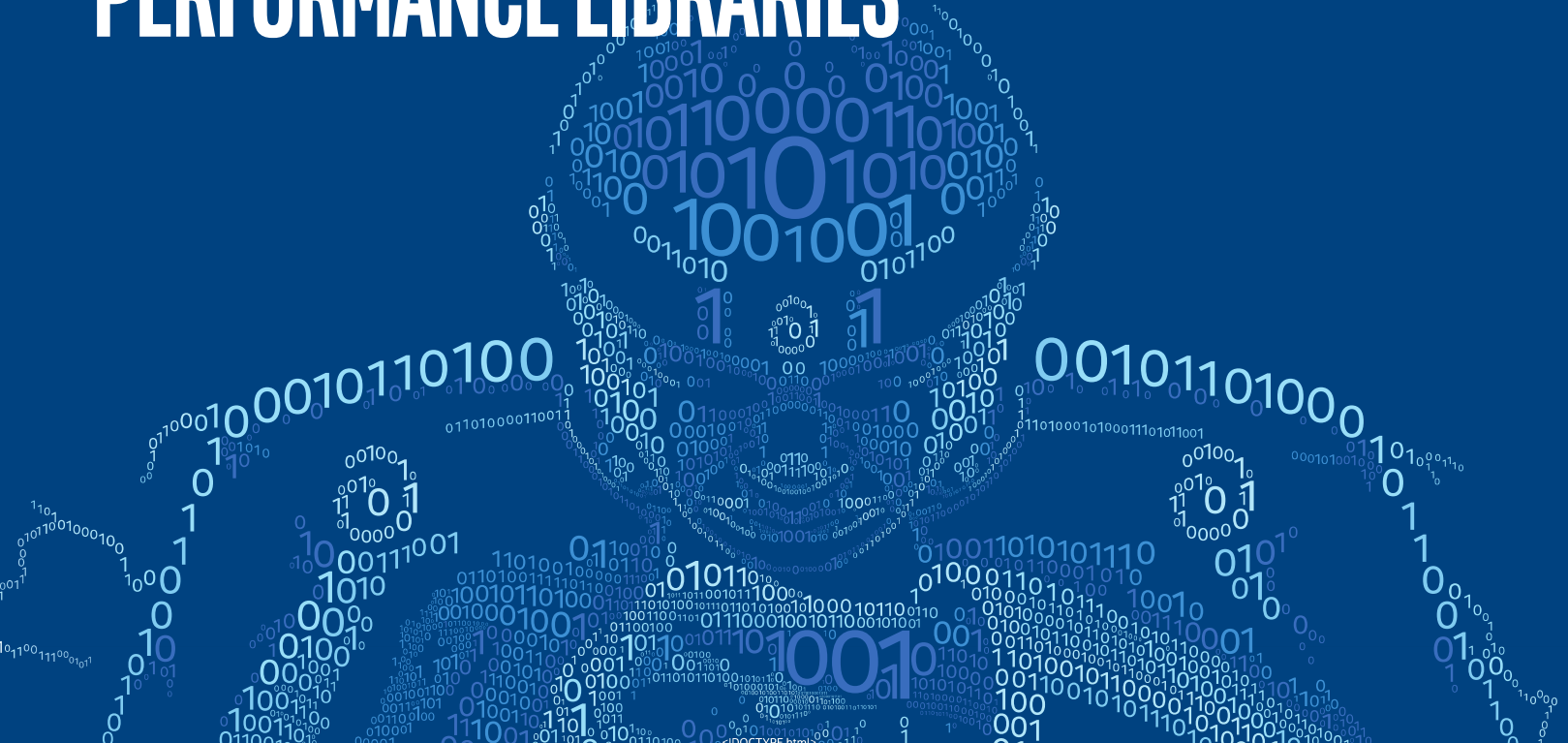
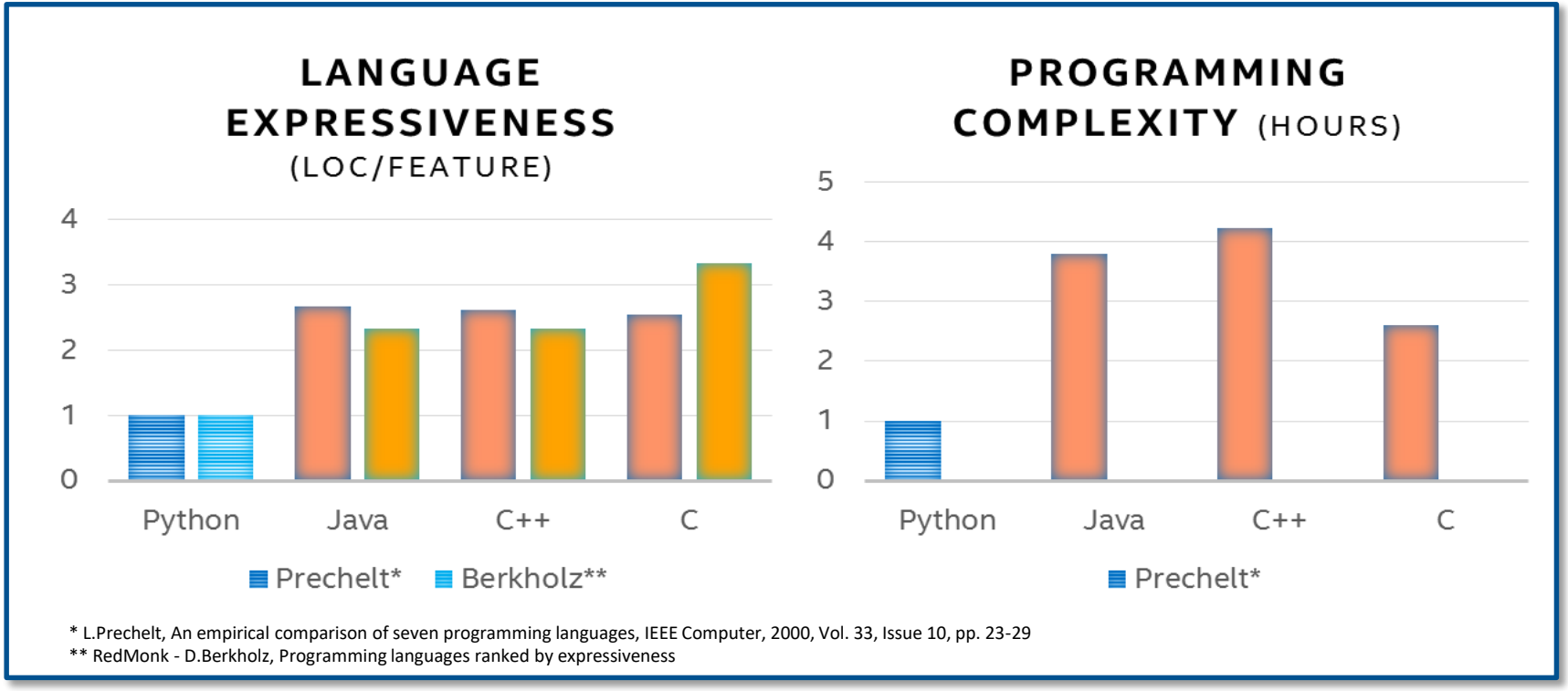
INTEL® DISTRIBUTION FOR PYTHON* И INTEL® PERFORMANCE LIBRARIES



intel Software

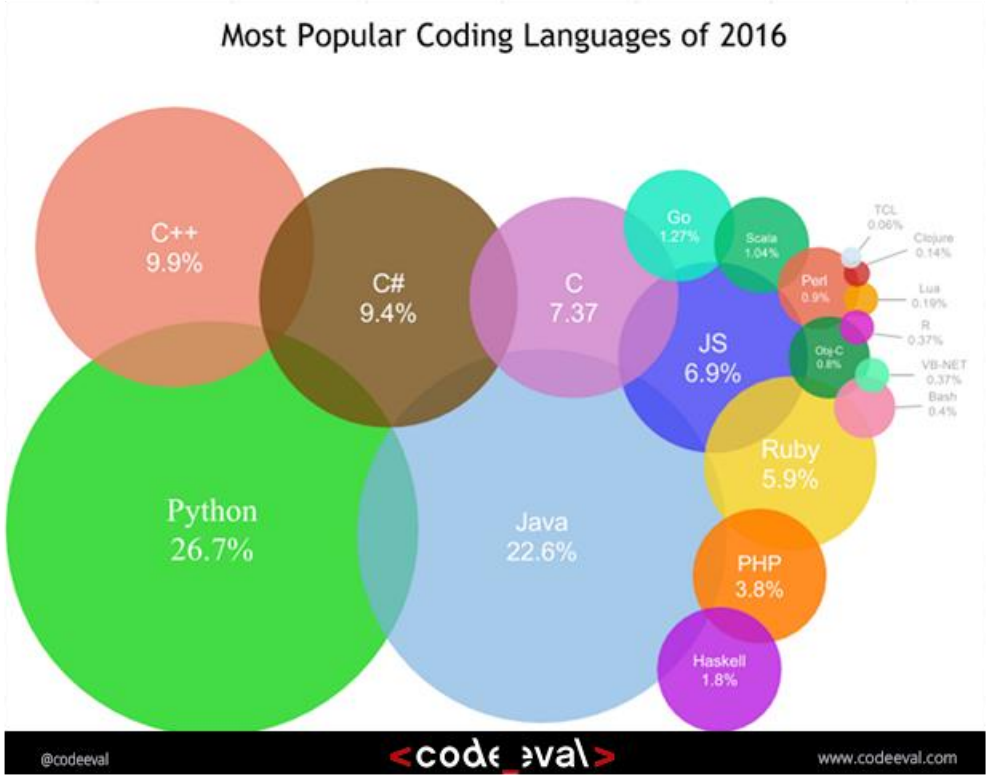
MOTIVATION



* L.Prechelt, An empirical comparison of seven programming languages, IEEE Computer, 2000, Vol. 33, Issue 10, pp. 23-29
** RedMonk - D.Berkholz, Programming languages ranked by expressiveness

ADOPTION OF PYTHON

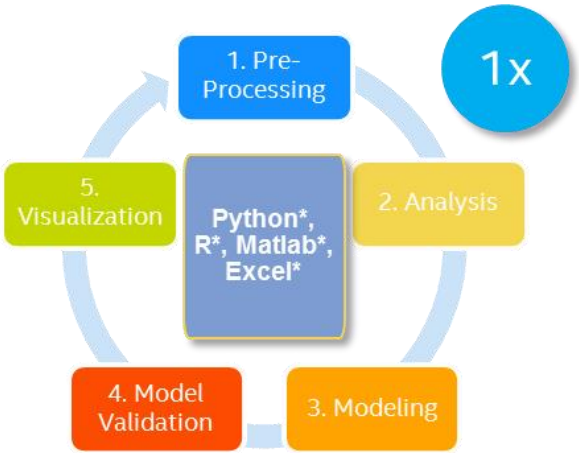
- continues to grow among domain specialists and developers for its productivity benefits



Source: CodeEval, Feb 2015

Workstation

Prototyping Development cost

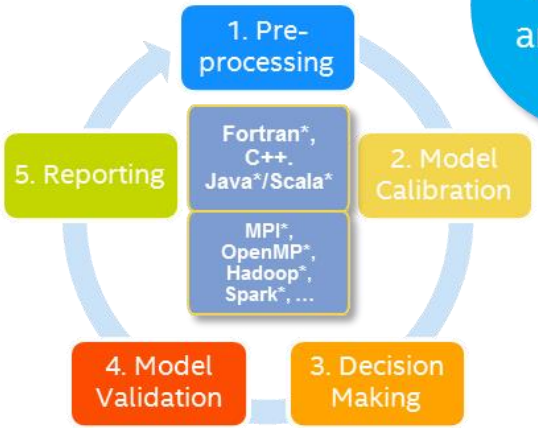


High migration costs

HPC/Big Data Cluster

Production Development cost

3-10x and more

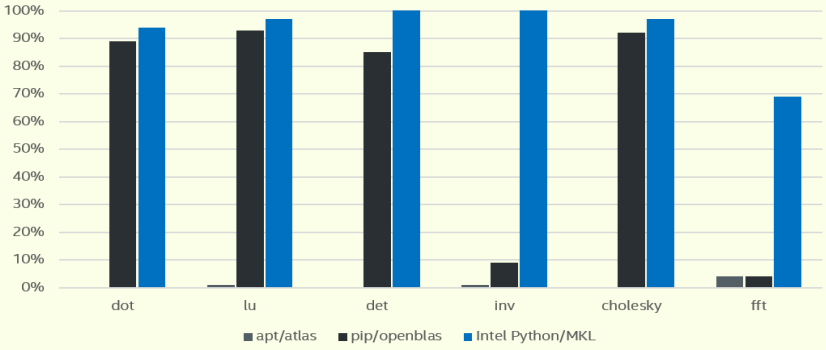


OUT-OF-THE-BOX PERFORMANCE WITH INTEL DISTRIBUTION FOR PYTHON

Mature AVX2 instructions based product

Intel® Xeon® Processors

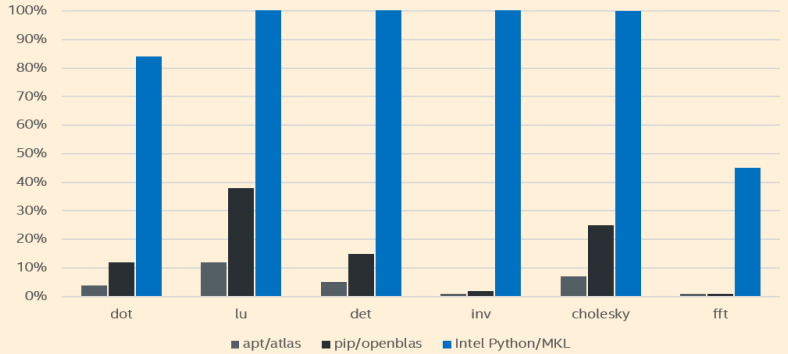
Python* Performance as a Percentage of C/Intel® MKL for Intel® Xeon® Processors, 32 Core (Higher is Better)



New AVX512 instructions based product

Intel® Xeon Phi™ Product Family

Python* Performance as a Percentage of C/Intel® MKL for Intel® Xeon Phi™ Product Family, 64 Core (Higher is Better)



Configuration Info: apt/atlas: installed with apt-get, Ubuntu 16.10, python 3.5.2, numpy 1.11.0, scipy 0.17.0; pip/openblas: installed with pip, Ubuntu 16.10, python 3.5.2, numpy 1.11.1, scipy 0.18.0; Intel Python: Intel Distribution for Python 2017;. Hardware: Xeon: Intel Xeon CPU E5-2698 v3 @ 2.30 GHz (2 sockets, 16 cores each, HT=off), 64 GB of RAM, 8 DIMMS of 8GB@2133MHz; Xeon Phi: Intel Intel® Xeon Phi™ CPU 7210 1.30 GHz, 96 GB of RAM, 6 DIMMS of 16GB@1200MHz

HIGHLIGHTS: INTEL® DISTRIBUTION FOR PYTHON* 2017

FOCUS ON ADVANCING PYTHON PERFORMANCE CLOSER TO NATIVE SPEEDS

Easy, out-of-the-box access to high performance Python

- Prebuilt, accelerated Distribution for numerical & scientific computing, data analytics, HPC. Optimized for IA
- Drop in replacement for your existing Python. No code changes required

Drive performance with multiple optimization techniques

- Accelerated NumPy/SciPy/scikit-learn with Intel® Math Kernel Library
- Data analytics with pyDAAL, Enhanced thread scheduling with TBB, Jupyter* notebook interface, Numba, Cython
- Scale easily with optimized mpi4py and Jupyter notebooks

Faster access to latest optimizations for Intel architecture

- Distribution and individual optimized packages available through conda and Anaconda Cloud
- Optimizations upstreamed back to main Python trunk

WHAT'S IN INTEL® DISTRIBUTION FOR PYTHON*? SCIPY-STACK + SELECTED BIGDATA/ML/HPC PACKAGES

Math/Compute

- Numpy
- Scipy
- pyDAAL
- Scikit-learn
- Numexpr
- Sympy
- Mpmath

Intel MKL
IPP
Intel DAAL
Intel Compiler

Parallelism/Performance

- TBB
- Mpi4py
- Numba
- Cython
- Pyzmq
- Distarray
- Pandas
- Pytables
- H5py

Intel TBB
Intel MPI
Intel Compiler

Productivity

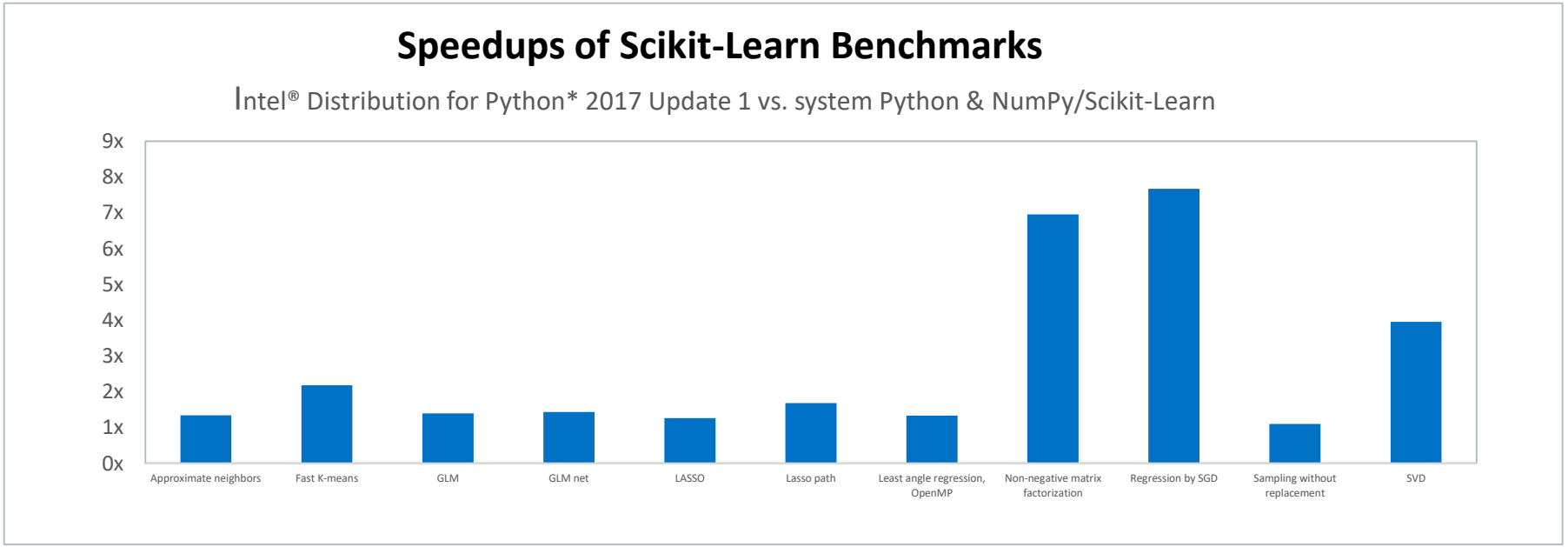
- Conda
- Pip
- Jupyter
- Notebook
- Matplotlib
- Nose/pytest/mock

Powered by ANACONDA Powered by Continuum Analytics

Misc

- Python 2.7/3.5
- Jinja2
- Pyyaml
- Tornado
- Llvmlite
- Six
- MarkupSafe
- Pytz
- Dateutil
- ...

SCIKIT-LEARN* OPTIMIZATIONS WITH INTEL® MKL



System info: 32x Intel® Xeon® CPU E5-2698 v3 @ 2.30GHz, disabled HT, 64GB RAM; Intel® Distribution for Python* 2017 Gold; Intel® MKL 2017.0.0; Ubuntu 14.04.4 LTS; Numpy 1.11.1; scikit-learn 0.17.1. See Optimization Notice.

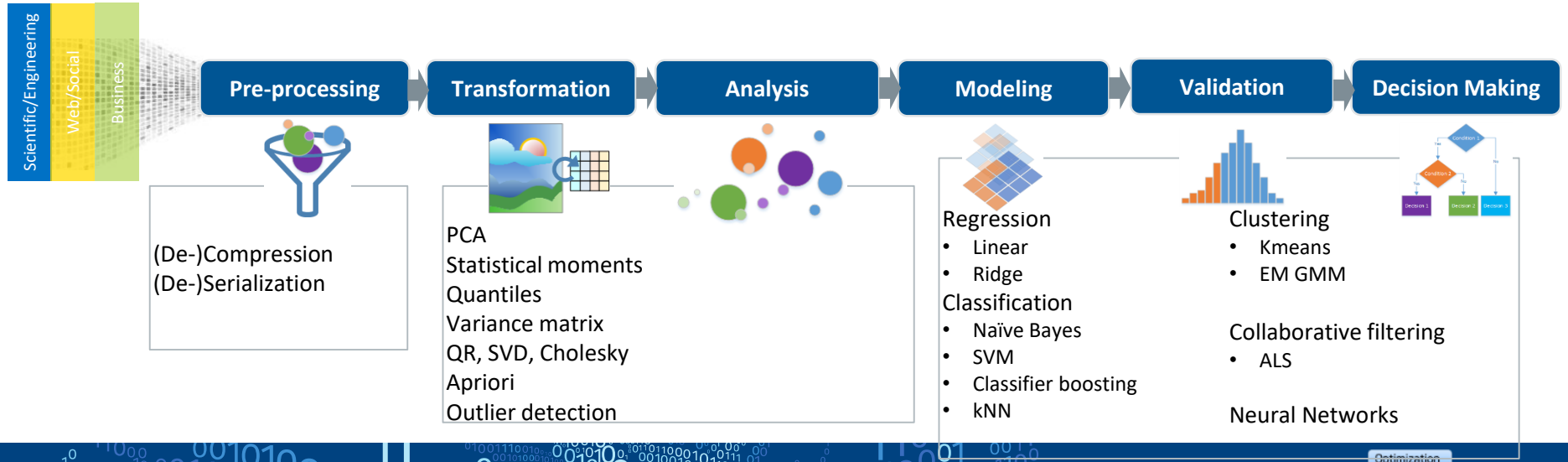
Software and workloads used in performance tests may have been optimized for performance only on Intel microprocessors. Performance tests, such as SYSmark and MobileMark, are measured using specific computer systems, components, software, operations and functions. Any change to any of those factors may 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. * Other brands and names are the property of their respective owners. Benchmark Source: Intel Corporation

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 .

INTEL® DAAL: HETEROGENEOUS ANALYTICS

Available also in open source:
https://software.intel.com/en-us/articles/opendaal

- Targets both data centers (Intel® Xeon® and Intel® Xeon Phi™) and edge-devices (Intel® Atom)
- Perform analysis close to data source (sensor/client/server) to optimize response latency, decrease network bandwidth utilization, and maximize security
- Offload data to server/cluster for complex and large-scale analytics



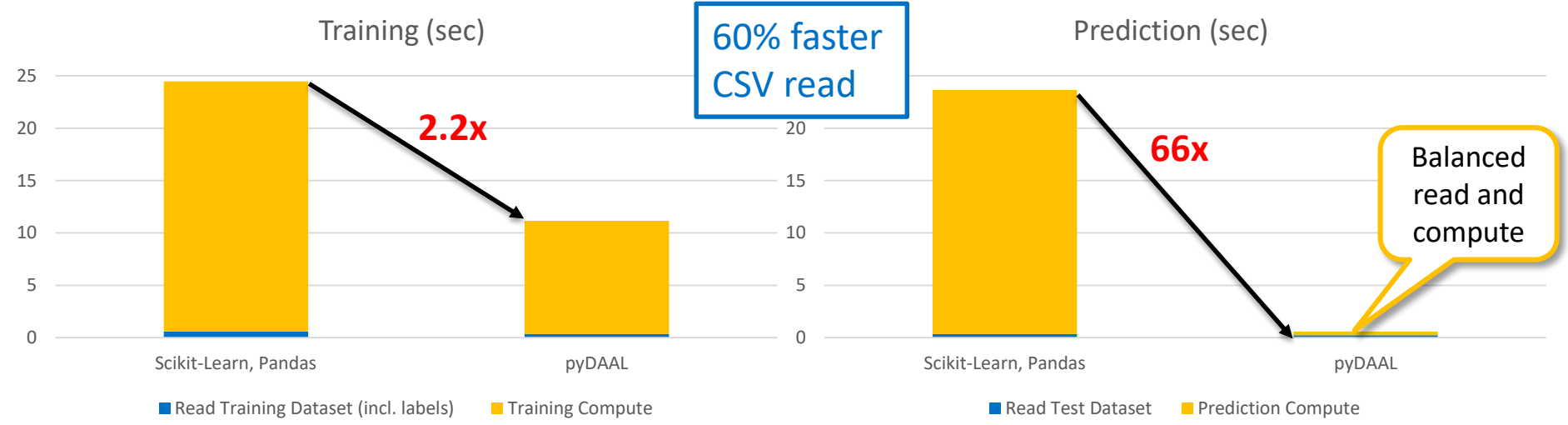
PERFORMANCE EXAMPLE : READ VS. COMPUTE

System Info: Intel(R) Xeon(R) CPU E5-2680 v3 @ 2.50GHz, 504GB, 2x24 cores, HT=on, OS RH7.2 x86_64, Intel Distribution for Python 2017 Update 1 (Python 3.5)

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. * Other brands and names are the property of their respective owners. Benchmark Source: Intel Corporation

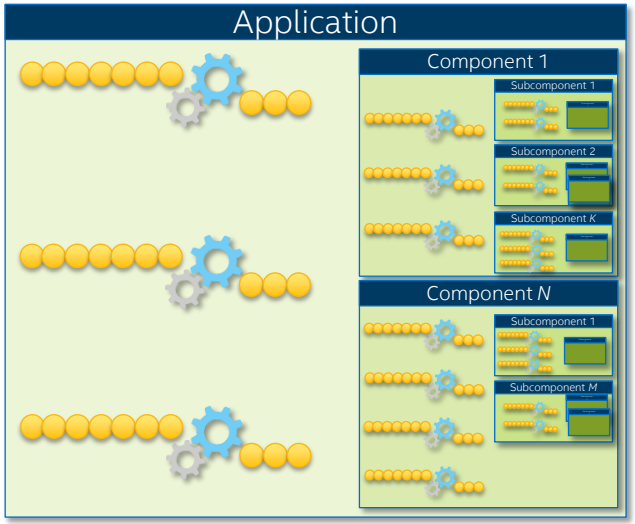
Optimization Notice: Intel's compilers may or may not optimize to the same degree for non-Intel microprocessors for optimizations that are not unique to Intel microprocessors. These optimizations include SSE2, SSE3, and SSSE3 instruction sets and other optimizations. Intel does not guarantee the availability, 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 .

- Algorithm: SVM Classification with RBF kernel
- Training dataset: CSV file (PCA-preprocessed MNIST, 40 principal components) *n*=42000, *p*=40
- Testing dataset: CSV file (PCA-preprocessed MNIST, 40 principal components) *n*=28000, *p*=40

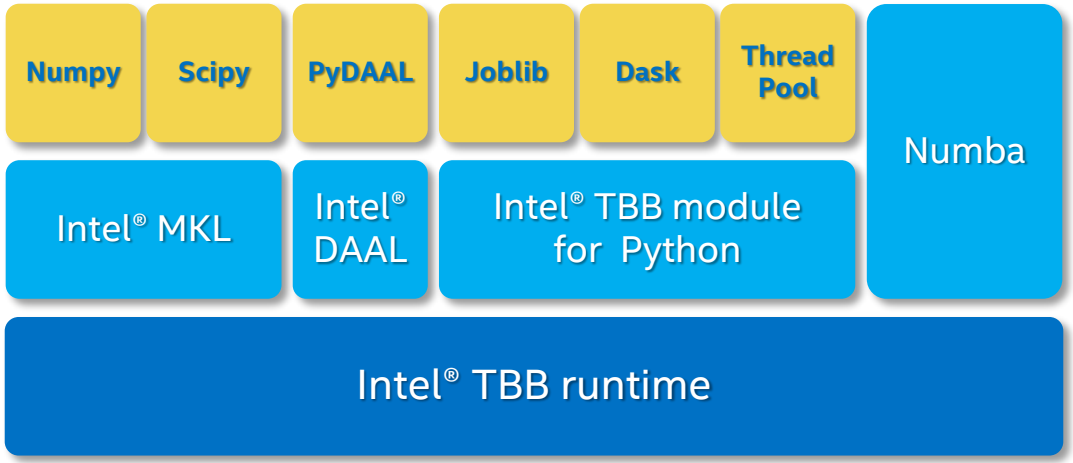


INTEL® TBB: PARALLELISM ORCHESTRATION IN PYTHON ECOSYSTEM

- Software components are built from smaller ones
- If each component is threaded there can be too much!
- Intel TBB dynamically balances thread loads and effectively manages oversubscription



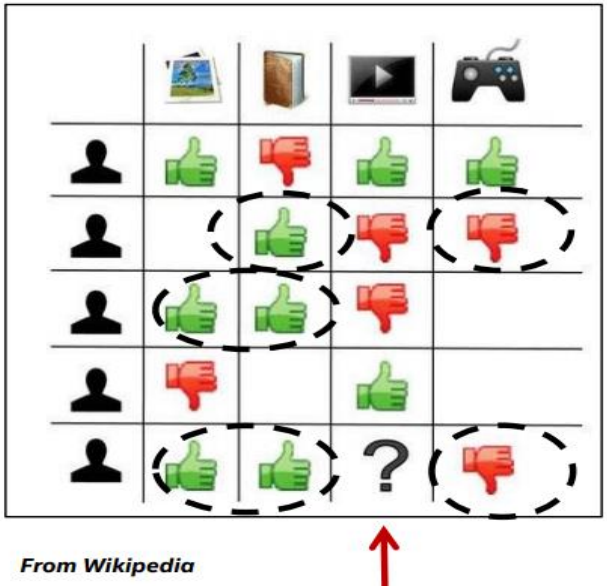

```
> python -m TBB application.py
```



COLLABORATIVE FILTERING

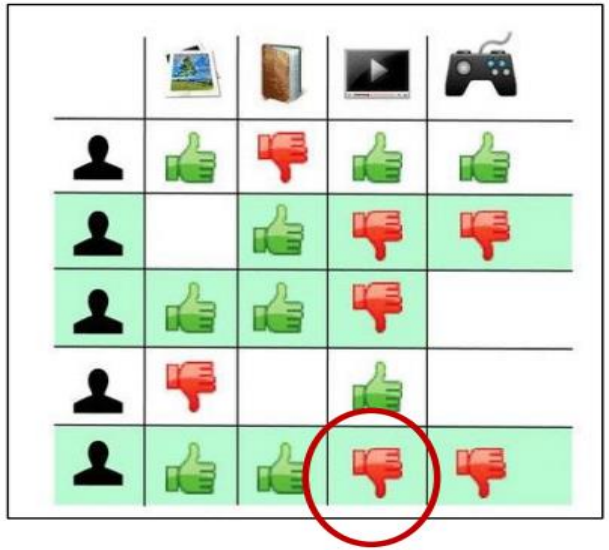
- Processes users' past behavior, their activities and ratings
- Predicts, what user might want to buy depending on his/her preferences

Collaborative Filtering



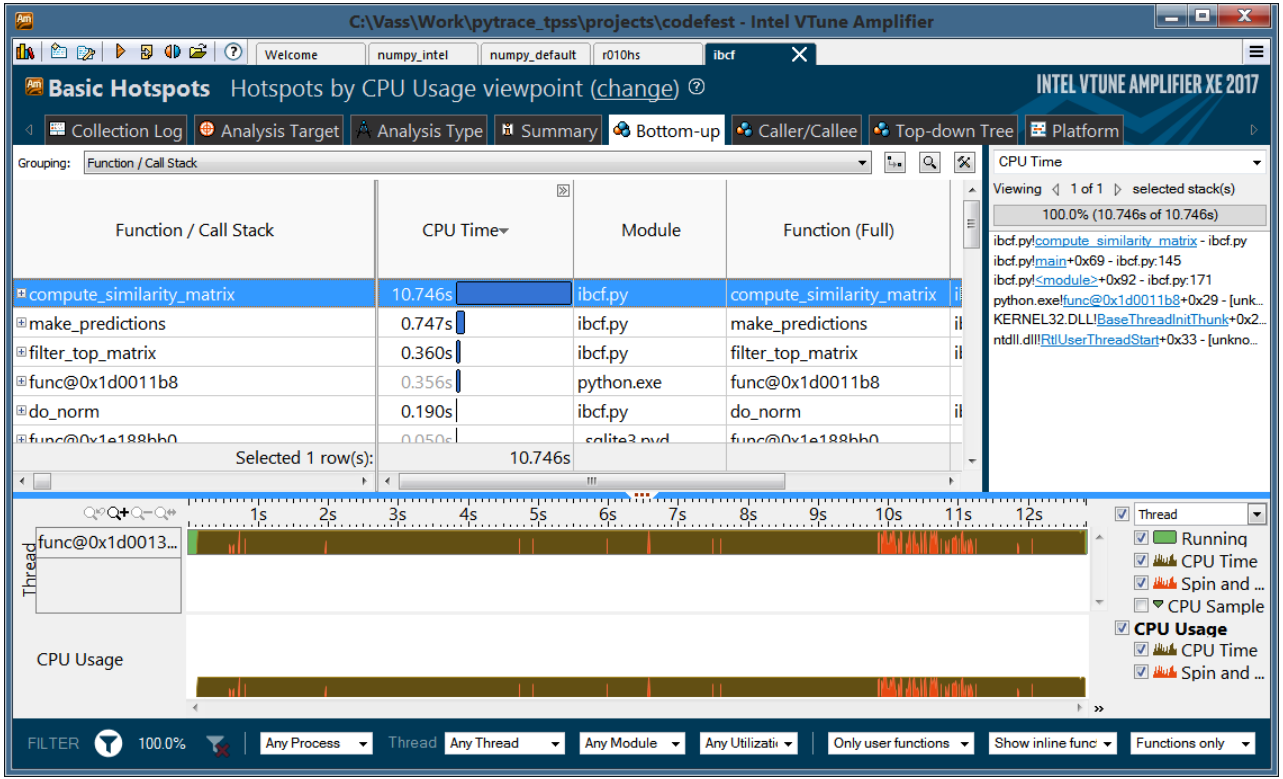
From Wikipedia

Similarities in users preferences (in Green) are used to predict ratings



TRAINING: PROFILING PURE PYTHON

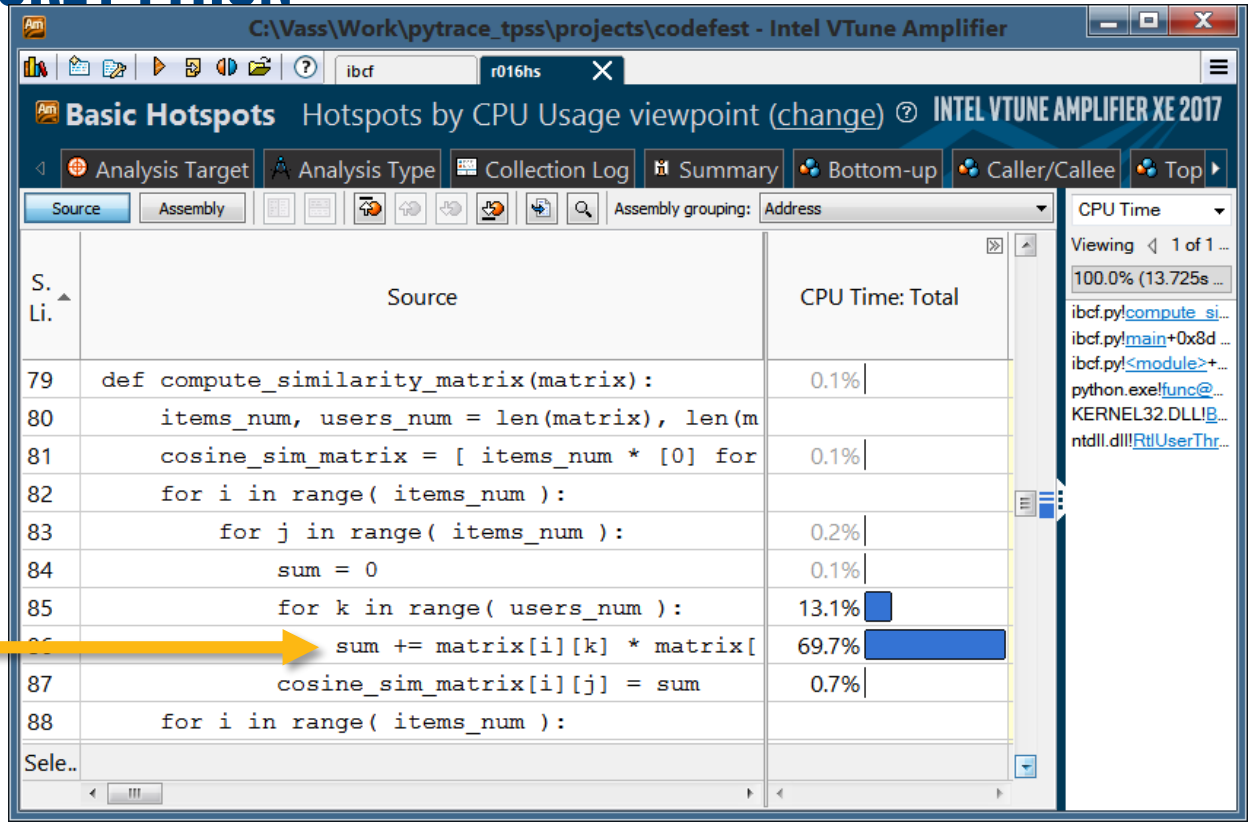
Items similarity assessment (similarity matrix computation) is the main hotspot



Configuration Info: - Versions: Red Hat Enterprise Linux* built Python*: Python 2.7.5 (default, Feb 11 2014), NumPy 1.7.1, SciPy 0.12.1, multiprocessing 0.70a1 built with gcc 4.8.2; Hardware: 24 CPUs (HT ON), 2 Sockets (6 cores/socket), 2 NUMA nodes, Intel(R) Xeon(R) X5680@3.33GHz, RAM 24GB, Operating System: Red Hat Enterprise Linux Server release 7.0 (Maipo)

TRAINING: PROFILING PURE PYTHON

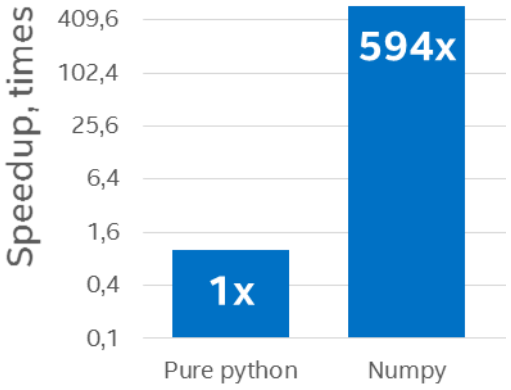
This loop is major bottleneck. Use appropriate technologies (NumPy/SciPy/Scikit-Learn or Cython/Numba) to accelerate

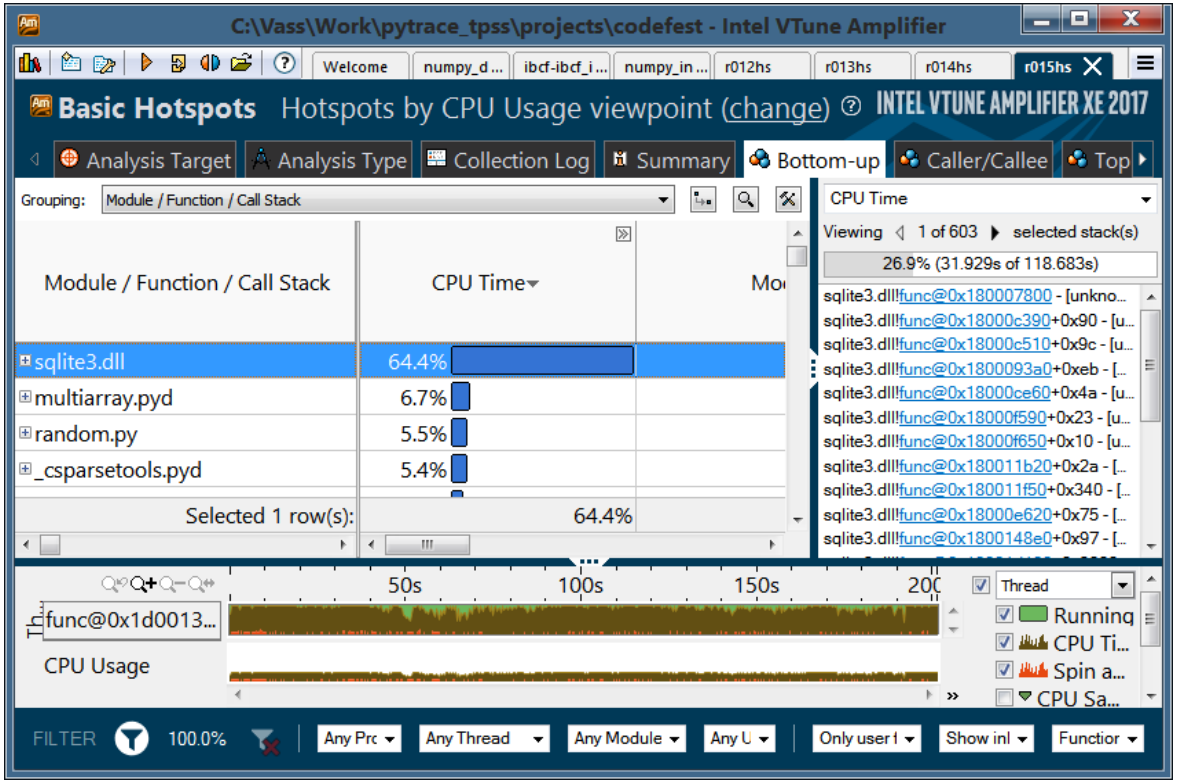


Configuration Info: - Versions: Red Hat Enterprise Linux* built Python*: Python 2.7.5 (default, Feb 11 2014), NumPy 1.7.1, SciPy 0.12.1, multiprocessing 0.70a1 built with gcc 4.8.2; Hardware: 24 CPUs (HT ON), 2 Sockets (6 cores/socket), 2 NUMA nodes, Intel(R) Xeon(R) X5680@3.33GHz, RAM 24GB, Operating System: Red Hat Enterprise Linux Server release 7.0 (Maipo)

TRAINING: PYTHON + NUMPY (MKL)

- Much faster!
- The most compute-intensive part takes ~5% of all the execution time

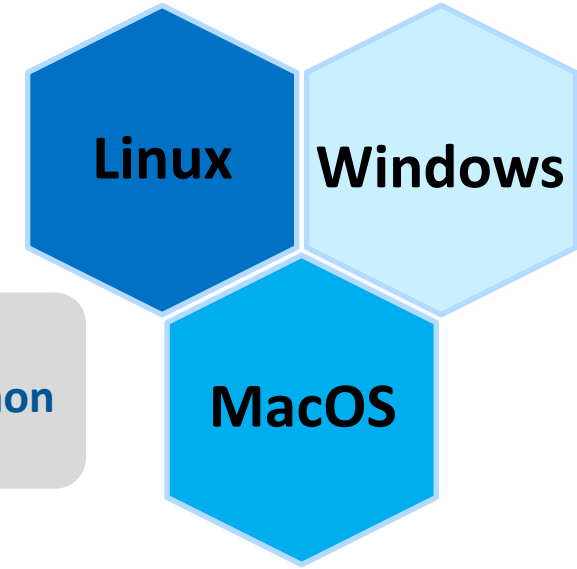




Configuration info: 96 CPUs (HT ON), 4 Sockets (12 cores/socket), 1 NUMA nodes, Intel(R) Xeon(R) E5-4657L v2@2.40GHz, RAM 64GB, Operating System: Fedora release 23 (Twenty Three)

INSTALLING INTEL® DISTRIBUTION FOR PYTHON* 2017

Stand-alone installer and on anaconda.org/intel

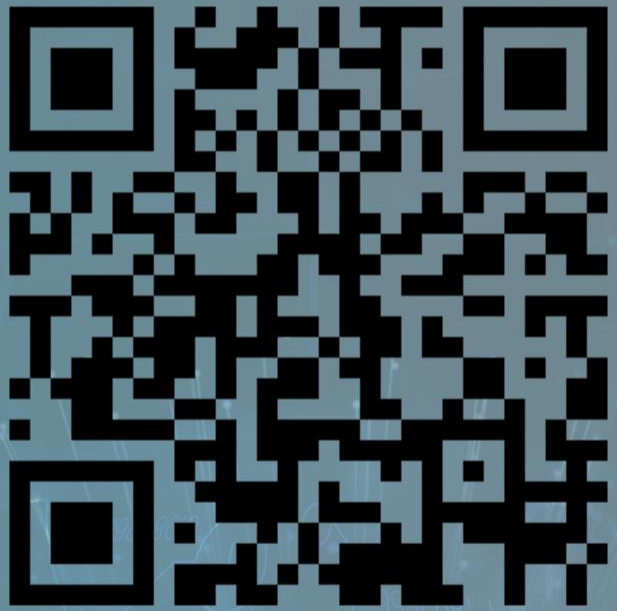


Download full installer from
https://software.intel.com/en-us/intel-distribution-for-python

or

```
➢ conda config --add channels intel
➢ conda create -n idp intelpython3_core python=3
➢ conda create -n idp intelpython3_full python=3
➢ source activate idp
```

INTEL® DISTRIBUTION FOR PYTHON*



https://software.intel.com/en-us/intel-distribution-for-python

LEGAL DISCLAIMER & OPTIMIZATION NOTICE

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