Estimating performance of Hadoop systems with deep learning

Daniele Grattarola

September 16, 2016

Contents

Abstract

In this paper I describe the application of deep learning to the estimation of the completion times of Hive queries in distributed Hadoop systems.

I show that the metric of interest is heavily correlated to some of the features used to describe the systems, and that good predictions can be obtained with a relatively simple deep neural network.

Results show that the model is able to generalize well enough on new data, both in leave-one-out cross validation and with more a strict validation on features values never seen in the training phase.

1 Introduction

The Apache Hadoop framework is an open-source software framework for distributed storage and distributed processing of very large data sets on computer clusters built from commodity hardware.

Apache Hive is a data warehouse infrastructure built on top of Hadoop for providing data summarization, query, and analysis. Hive gives an SQL-like interface to query data stored in various databases and file systems that integrate with Hadoop. [3]

Due to the possibly great complexity of the distributed systems on which Hive queries are run, and enormous sizes of modern datawarehouses, it is often of interest to have insights on the performances obtained on specific types of queries.

In this paper, I considered as main performance indicator the completion times of eight types of query run on different hardware configurations, with a particular focus on the number of cores (nCores) feature.

2 Previous work

This paper was conceived as an extension of a similar work conducted by A. Battistello and P. Ferretti with Support Vector Machines and standard linear regression [1].

3 Problem formulation

3.1 Datasets

The data used to fit the models consisted in eight different datasets, each associated to a specific Hive query run on different hardware configurations.

The queries were divided into *simple* (queries R1, R2, R3, R4, and R5) and *complex* (queries Q2, Q3, and Q4) and of each query was recorded the completion time on different systems.

Systems were described with different features based on the type of query:

- R queries: the features considered were the number of map and reduce threads, average and maximum map time, average and maximum reduce time, average and maximum shuffle time, average and maximum number of bytes sent throught the network, number of users, size of the dataset against which the queries were run, and number of available cores
- Q queries: the considered features were the same as the R queries, but with the added map, reduce and shuffling times (average and maximum) of the intermediate nodes of the execution DAG.

3.2 Data analysis

In order to better understand the structure of the data and the internal dependencies between the features, I computed some additional information and used it to preprocess the data fed to the model.

For each dataset, the following was computed:

1. Correlation matrix between all features, including the target values
2. Projection of the feature space over the first principal component, using Principal Component Analysis to determine the dimensions which explained the most variance in the data
3. Projection of the feature space over the first two principal components, using Principal Component Analysis to determine the dimensions which explained the most variance in the data
4. Plot of the nCores feature against the complTime feature

The results of these operations are listed below.

3.2.1 Dataset R1

It is possible to see that the two most important features in this dataset are the maximum shuffle time and the number of cores, which have almost maximum direct and inverse correlation respectively with the target value.

PCA analisys shows that the points lie on an almost linear surface in the 3D PC space.

Figure 1: Correlation matrix R1

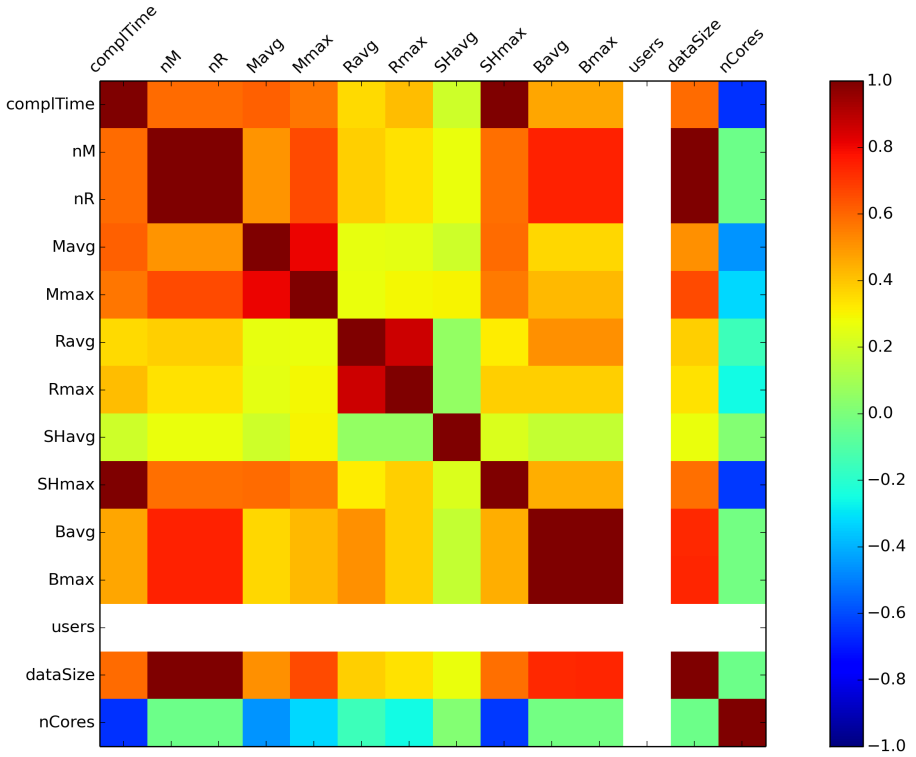


Figure 2: 2D PCA R1

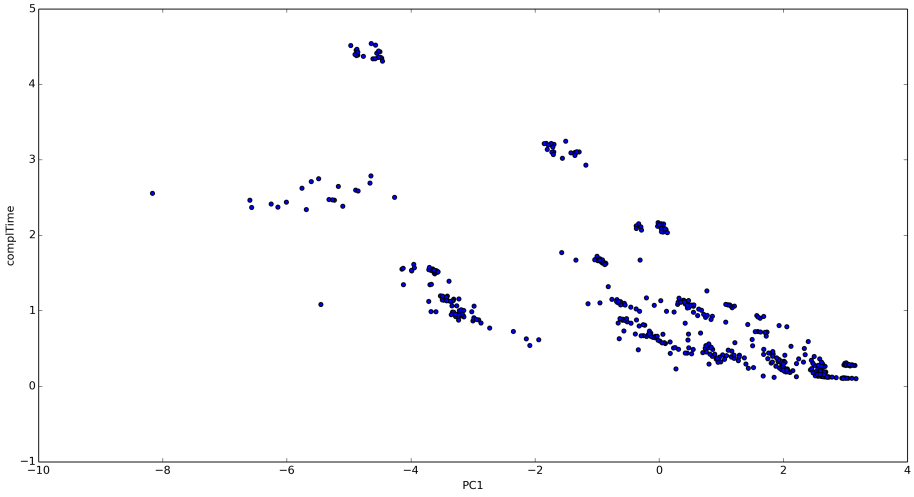


Figure 3: 3D PCA R1

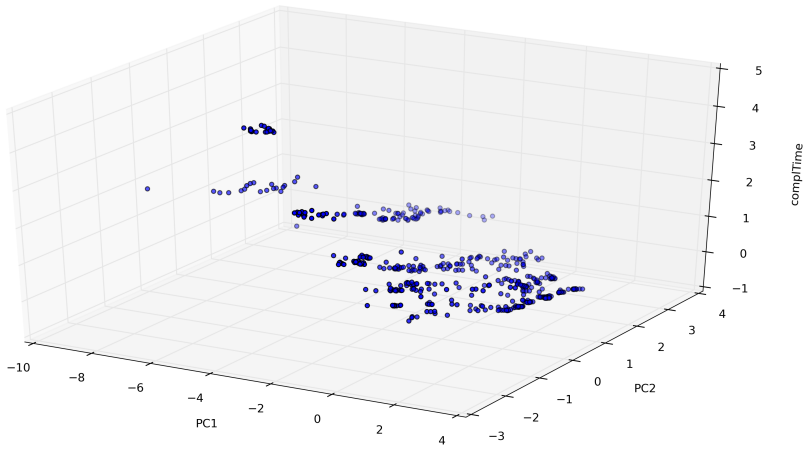
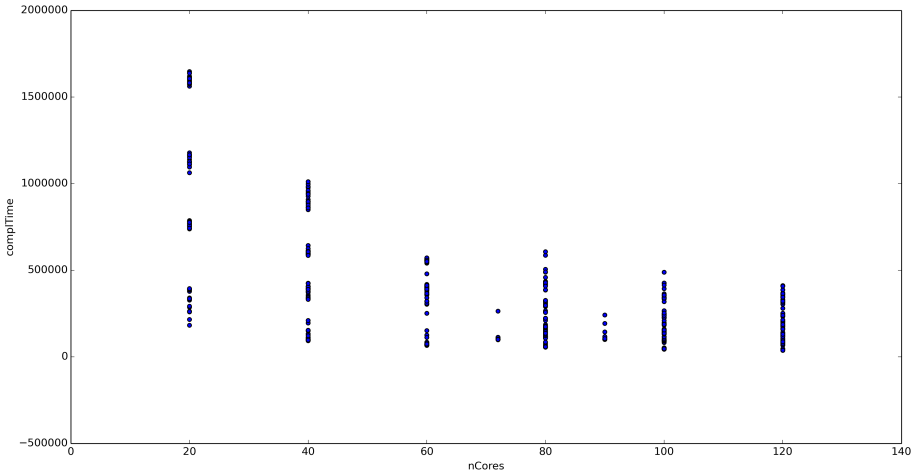


Figure 4: nCores vs. complTime R1



3.2.2 Dataset R2

It is possible to see that the two most important features in this dataset are related to the map time, whereas the number of cores becomes less relevant.

We can see that the data are alredy clearly structured in the 2D PC space.

Figure 5: Correlation matrix R2

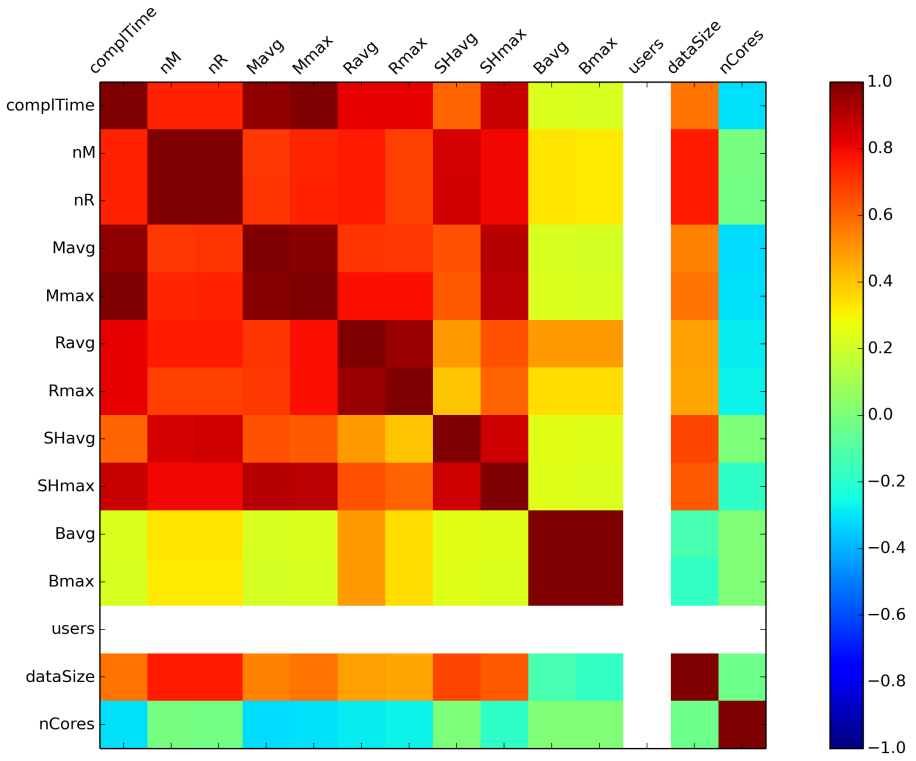


Figure 6: 2D PCA R2

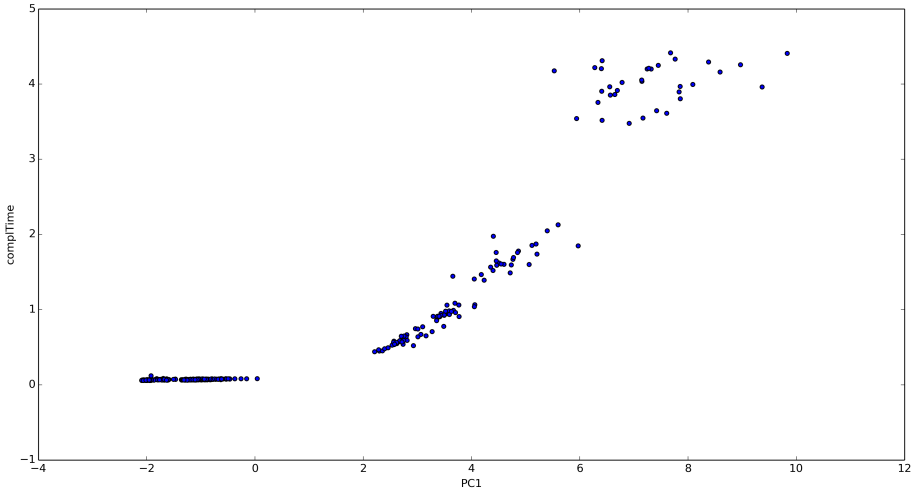


Figure 7: 3D PCA R2

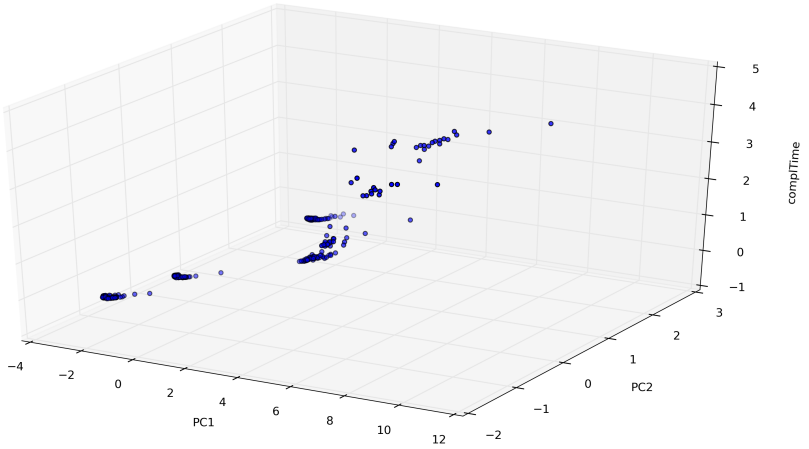
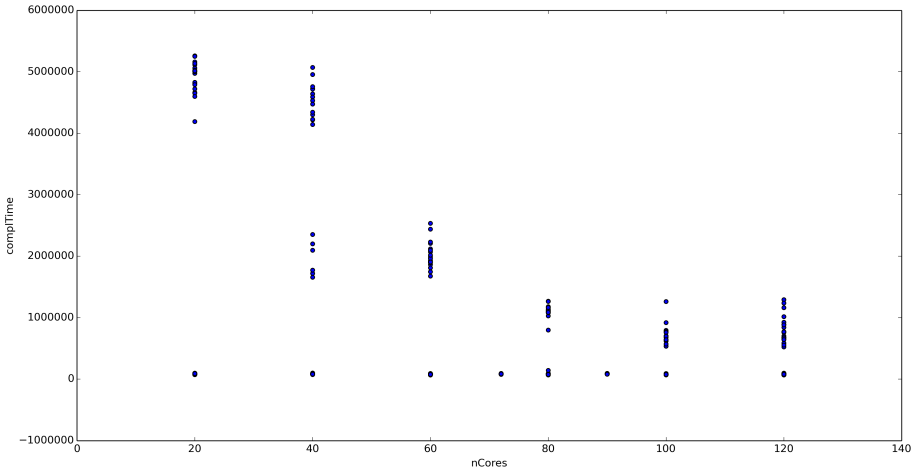


Figure 8: nCores vs. complTime R2



3.2.3 Dataset R3

This dataset has a structure similar to R1, with maximum shuffle time and number of cores being the most correlated features to the target values and point which lie on a surface in the 3D PC space.

Figure 9: Correlation matrix R3

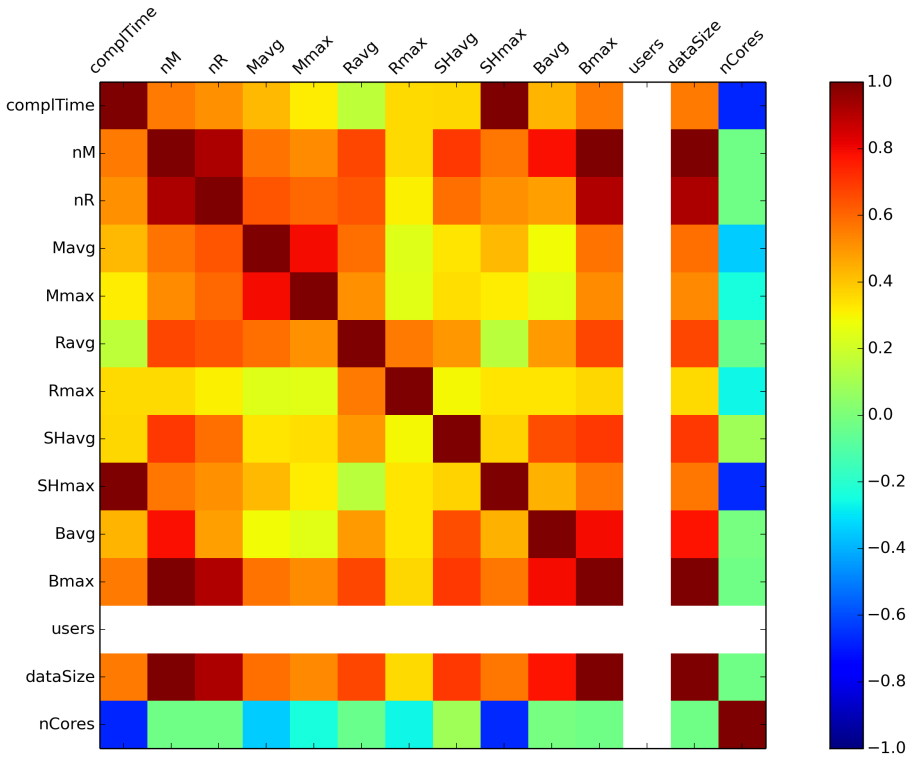


Figure 10: 2D PCA R3

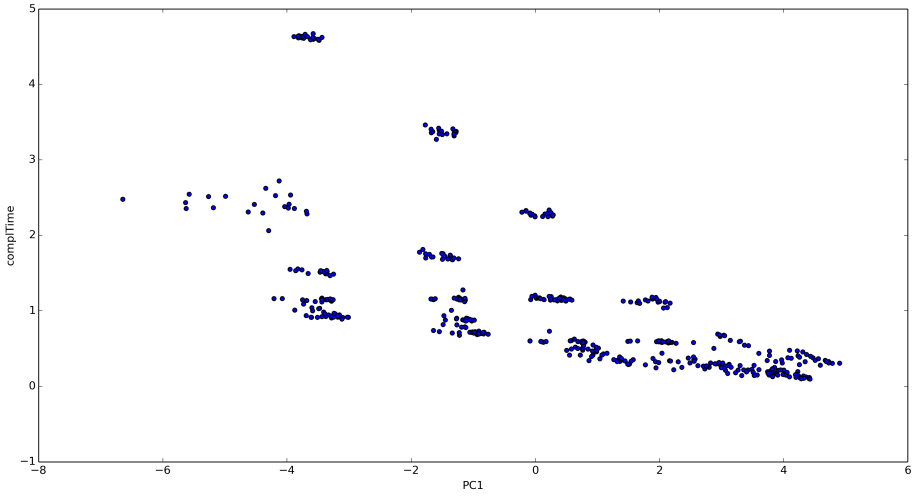


Figure 11: 3D PCA R3

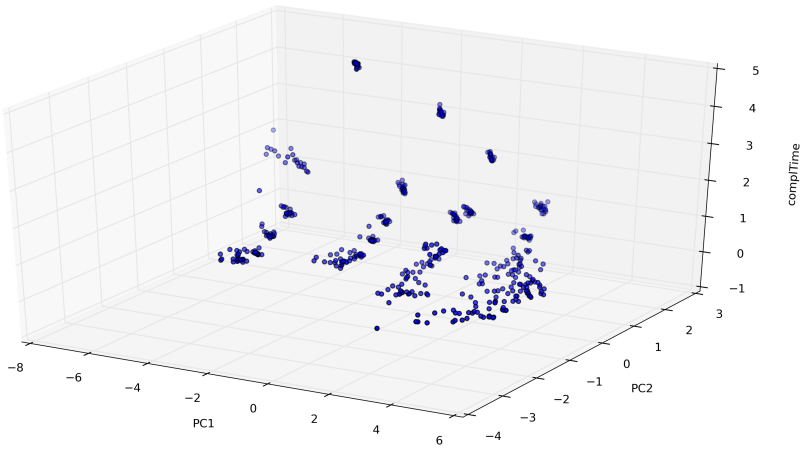
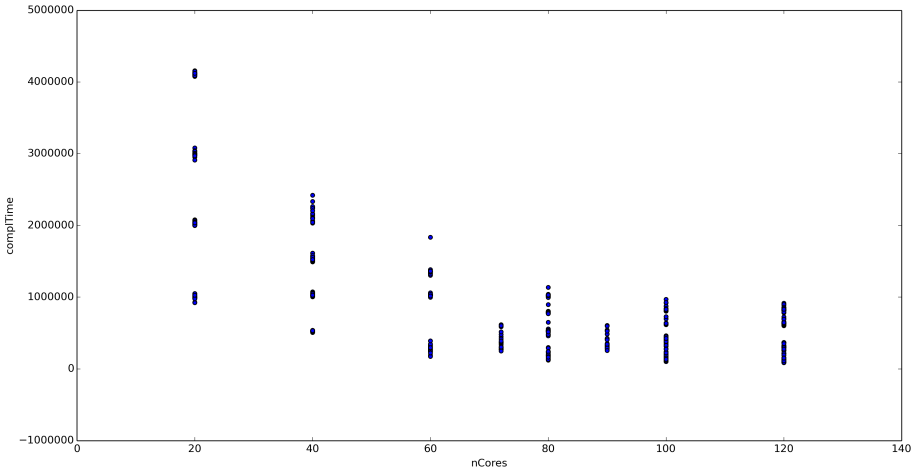


Figure 12: nCores vs. complTime R3



3.2.4 Dataset R4

We can see that the number of cores plays a non-noticeable role in this dataset, with the shuffle times being the most important.

Data show a clear structure in both the PC projections.

Figure 13: Correlation matrix R4

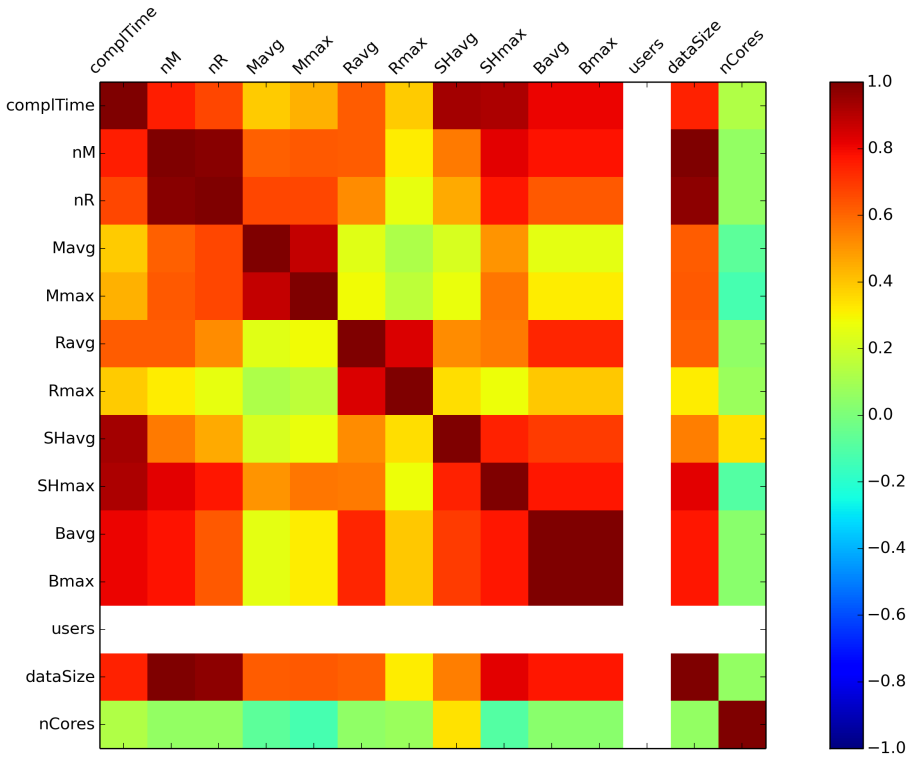


Figure 14: 2D PCA R4

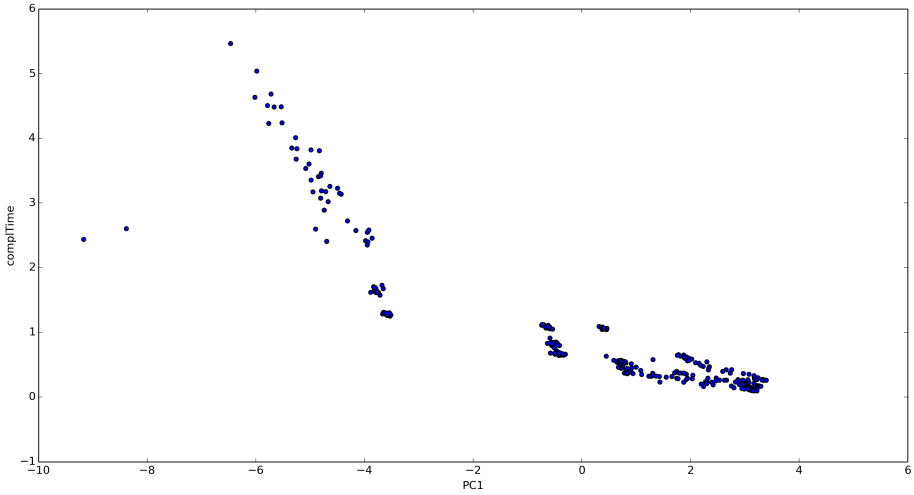


Figure 15: 3D PCA R4

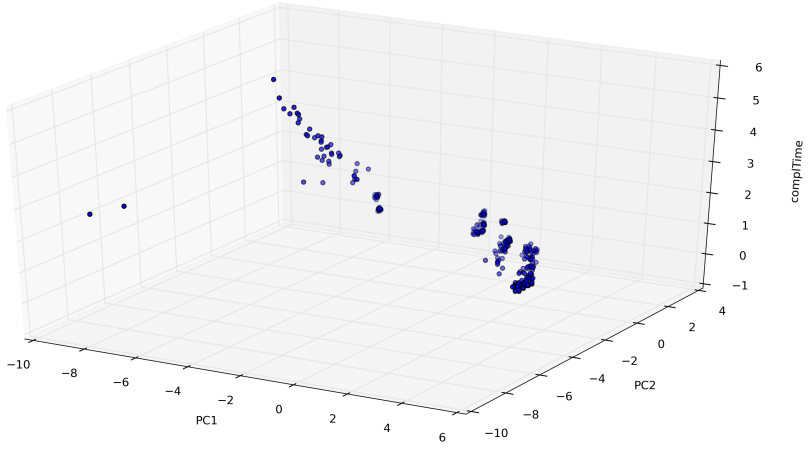
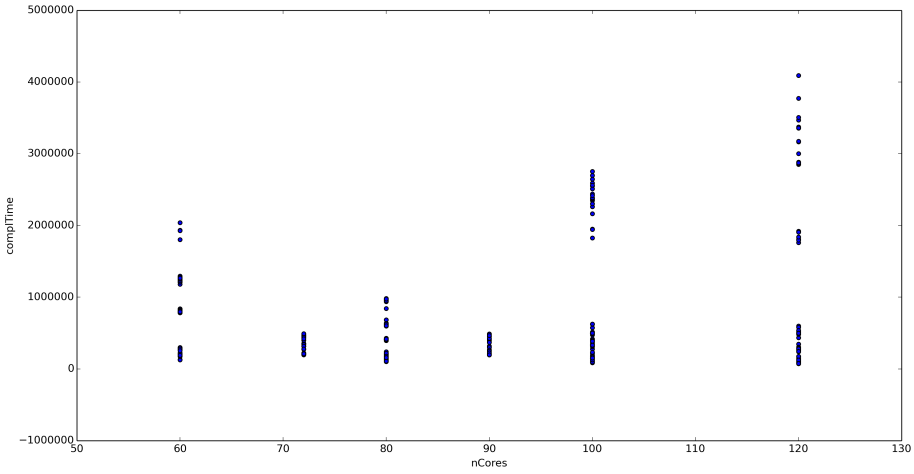


Figure 16: nCores vs. complTime R4



3.2.5 Dataset R5

While there's no clear "winner" like in the other datasets, here the correlation matrix shows that map times and bandwidth are more correlated to the target values than the other features. Once again, we see that the number of cores has low correlation to the target values.

Here, too, PCA highlights almost linear dependence in the data.

Figure 17: Correlation matrix R5

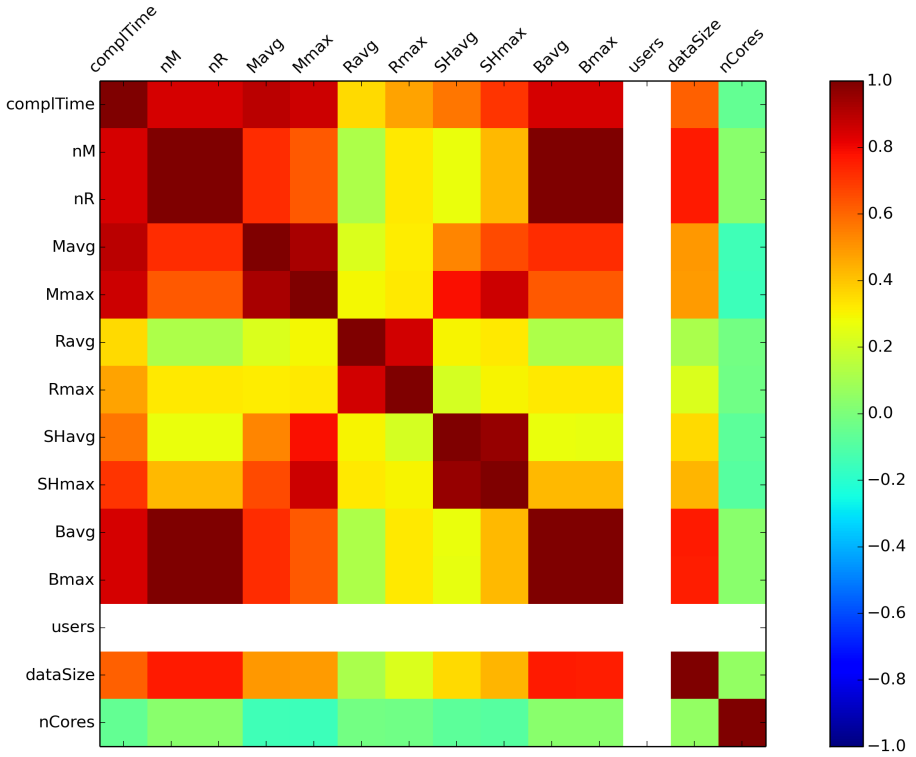


Figure 18: 2D PCA R5

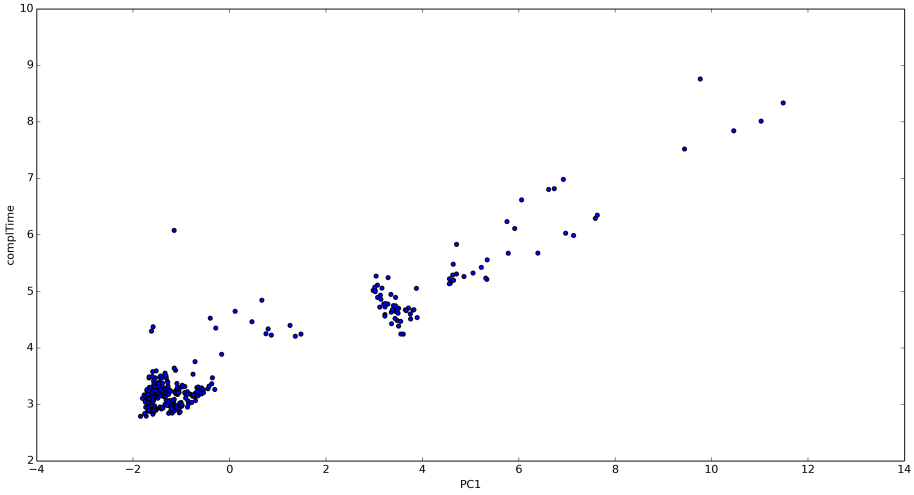


Figure 19: 3D PCA R5

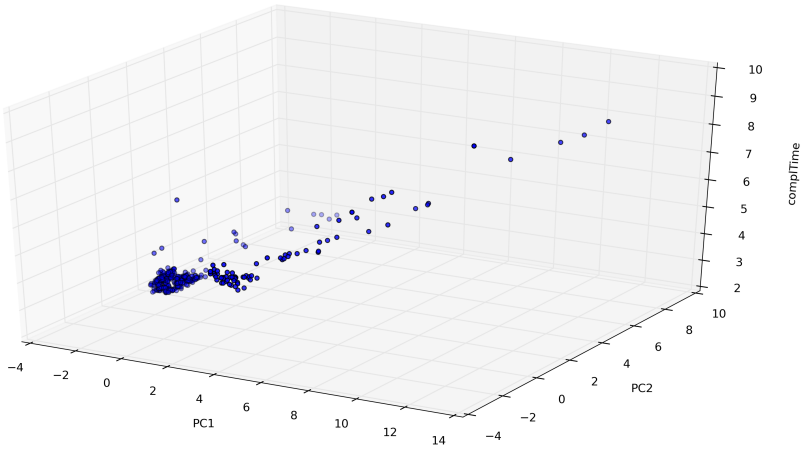
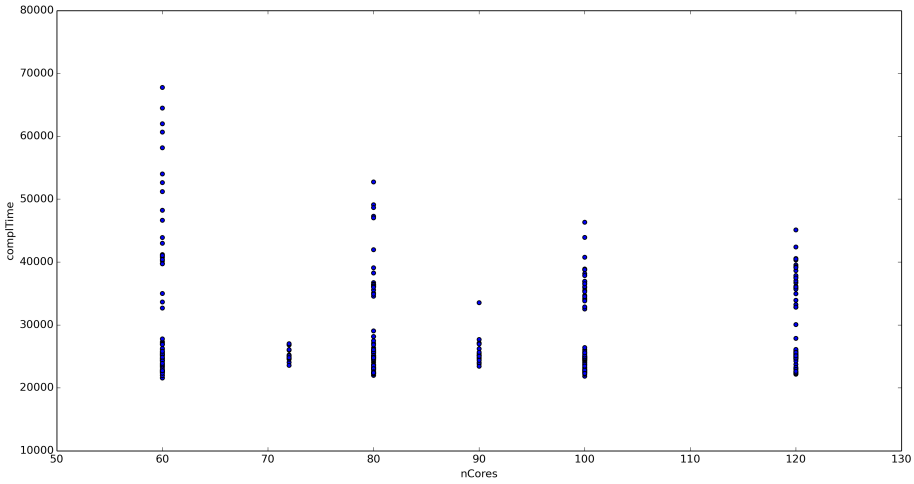


Figure 20: nCores vs. complTime R5



3.2.6 Dataset Q2

As the first complex query approached in this analysis, we expected to see strong correlation in more features, but it's possible to see how the biggest role are played by the first map task and by the number of cores.

Once again, there is clear structure in both the 2D and 3D PC spaces.

Figure 21: Correlation matrix Q2

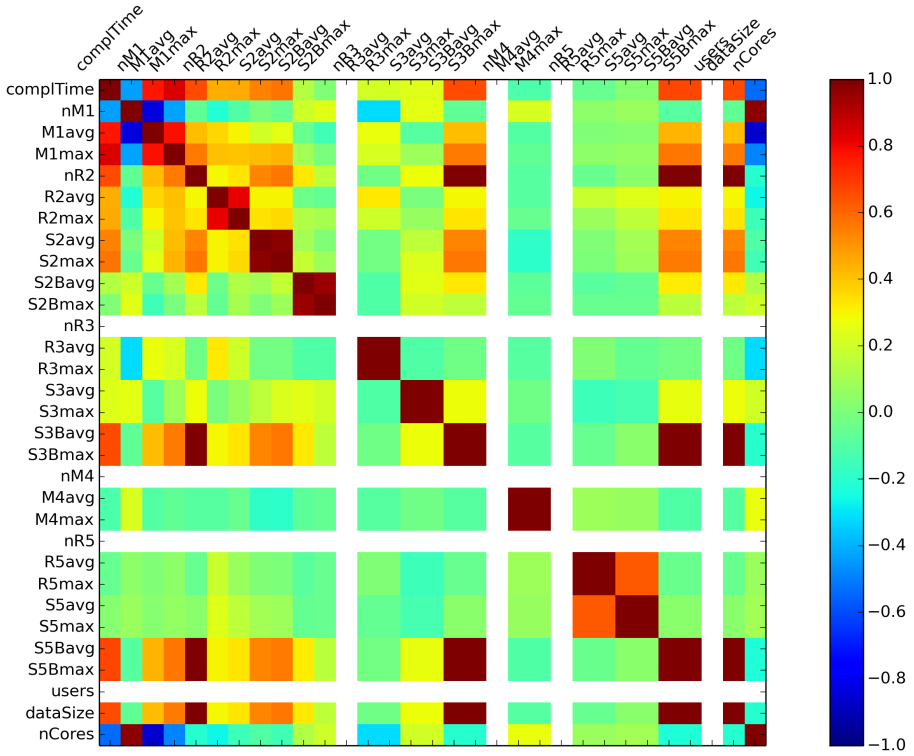


Figure 22: 2D PCA Q2

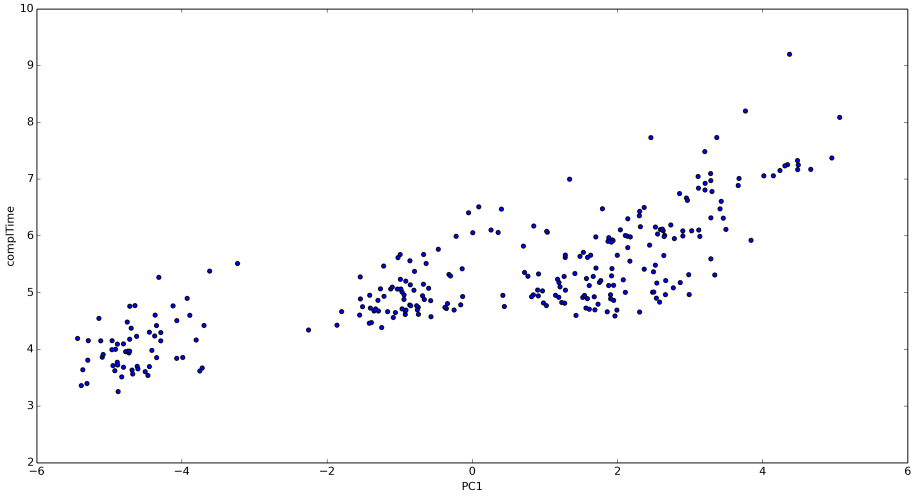


Figure 23: 3D PCA Q2

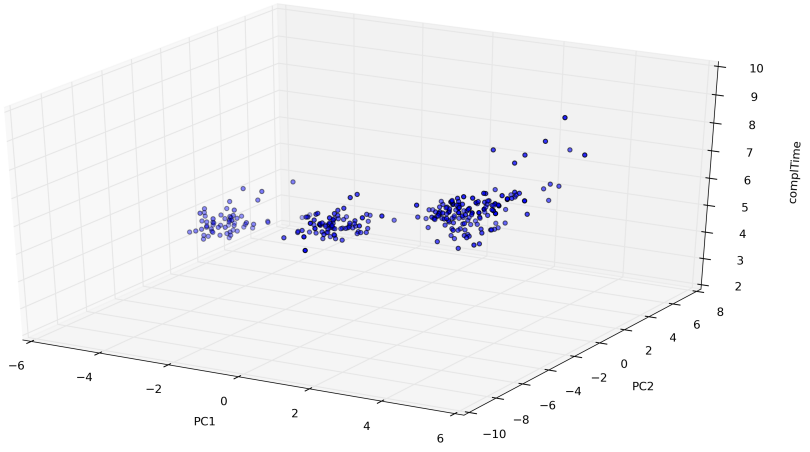
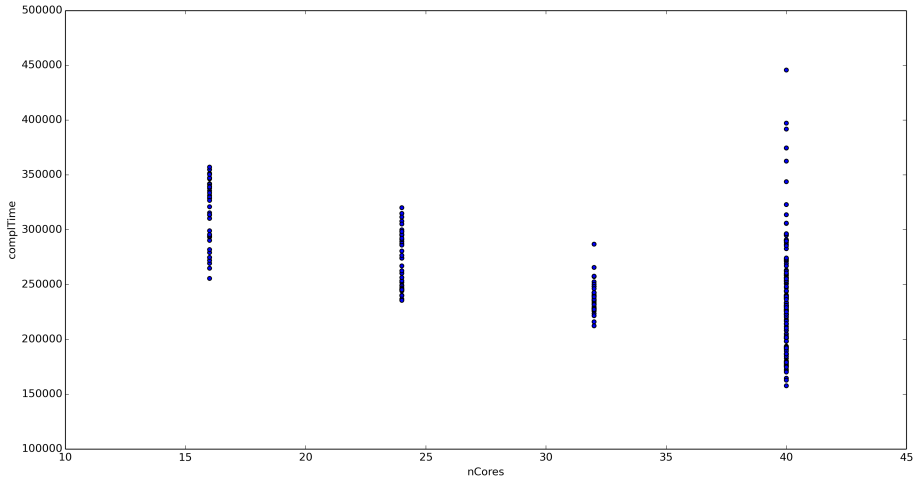


Figure 24: nCores vs. complTime Q2



3.2.7 Dataset Q3

Correlation analysis of this dataset shows strong inverse correlation to the available bandwidth and number of cores and, oddly enough, to the number of maps.

This is also the most noisy dataset, with no clear structure emerging from PCA except from a clusterization of points in the higher and lower regions of the domain.

Figure 25: Correlation matrix Q3

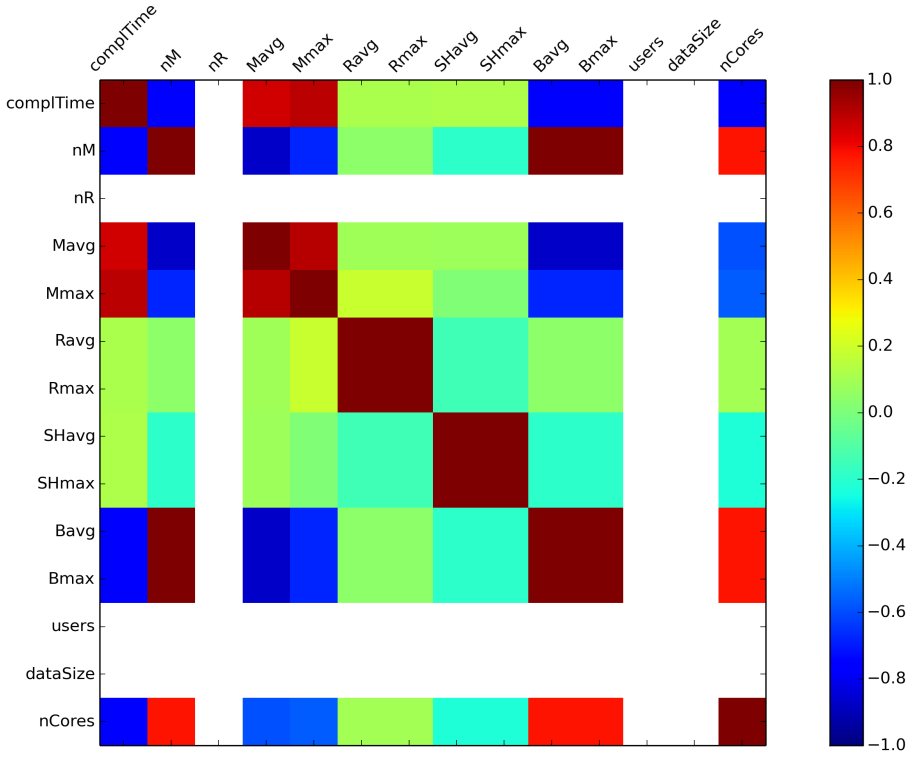


Figure 26: 2D PCA Q3

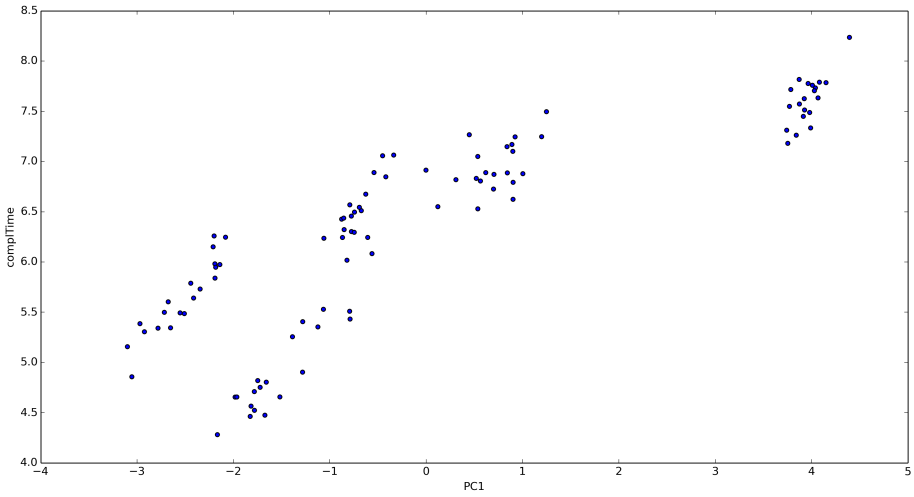


Figure 27: 3D PCA Q3

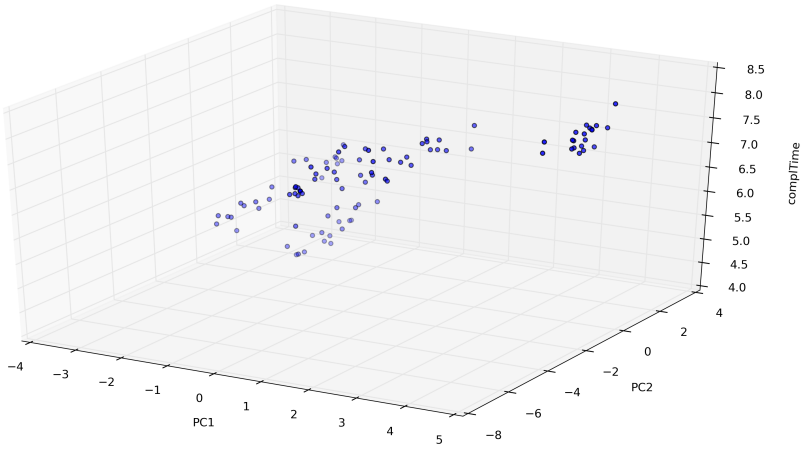
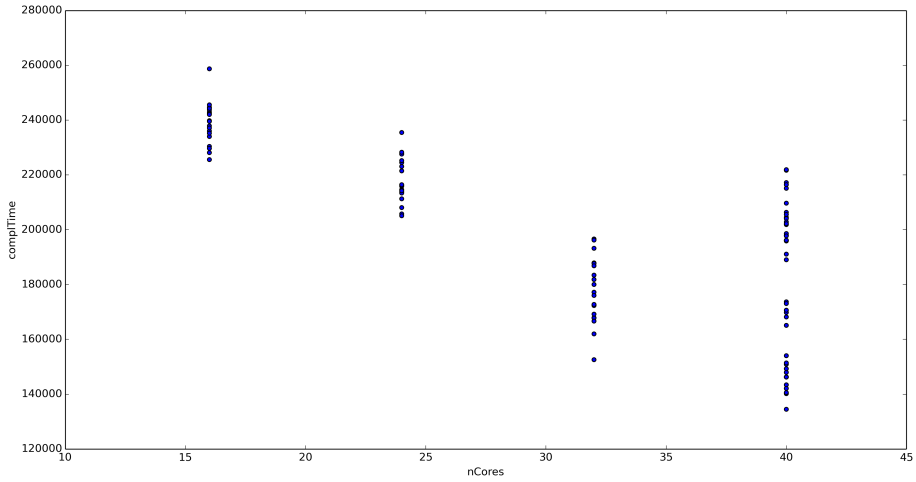


Figure 28: nCores vs. complTime Q3



3.2.8 Dataset Q4

This dataset also presents itself with odd inverse correlations, and no strong positive correlations.

Data have linear structure in the 2D space and clusters emerge in the 3D PC space.

Figure 29: Correlation matrix Q4

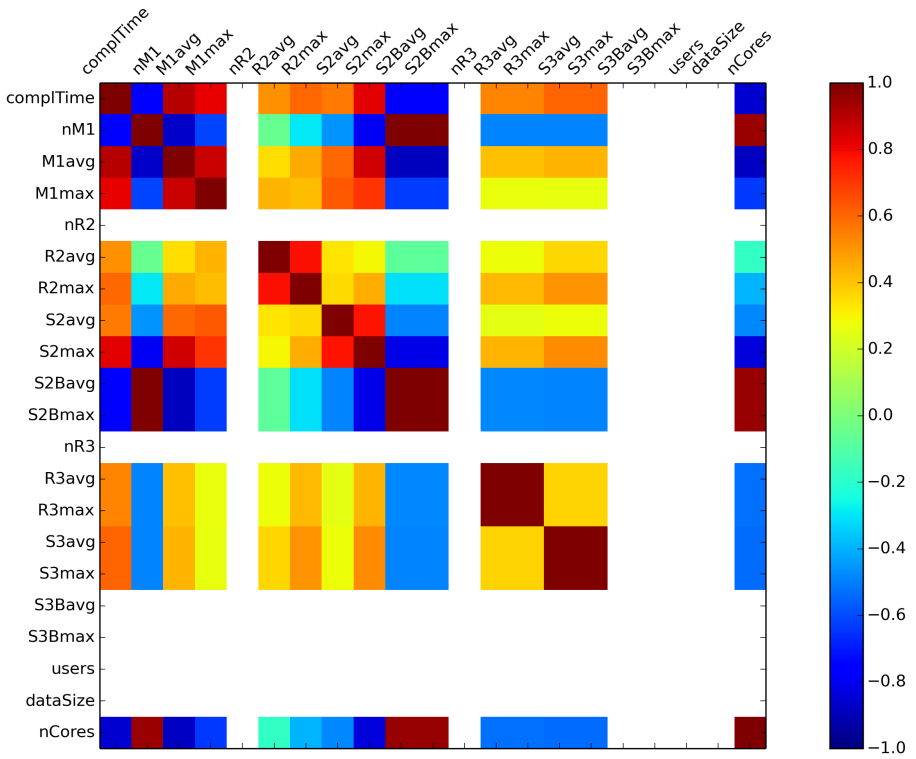


Figure 30: 2D PCA Q4

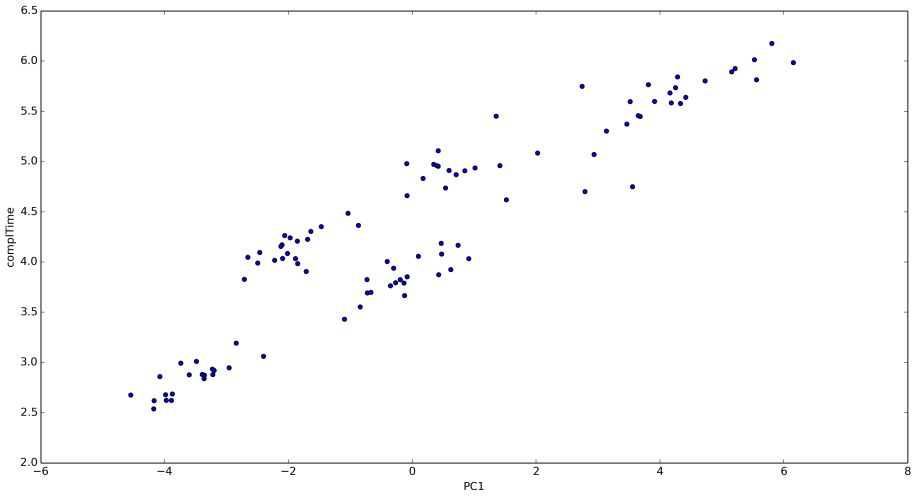


Figure 31: 3D PCA Q4

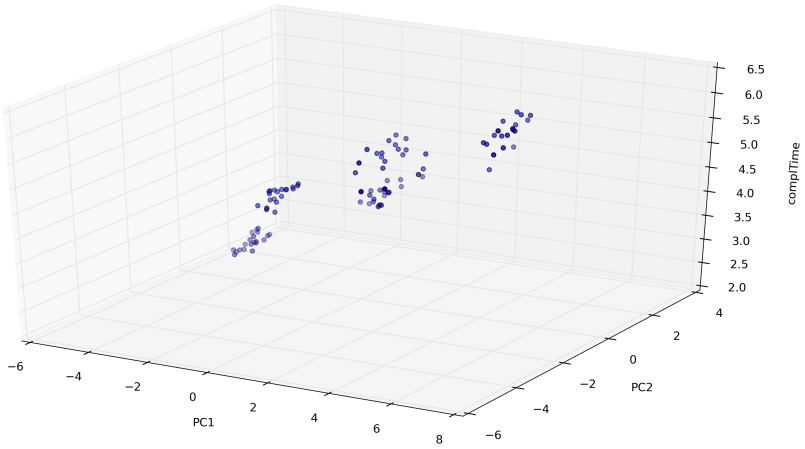
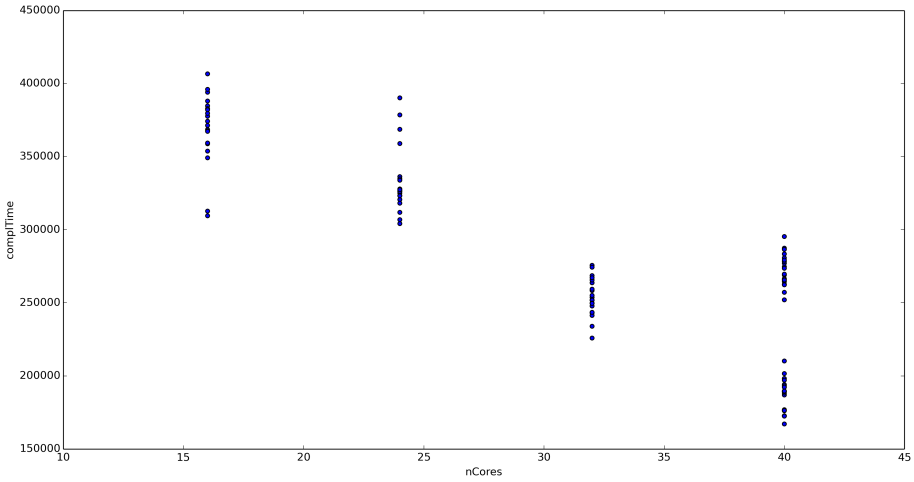


Figure 32: nCores vs. complTime Q4

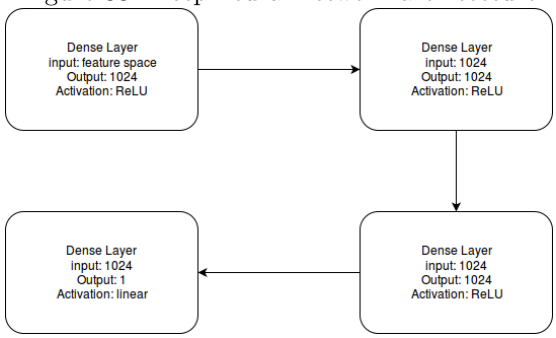


4 Methodology

In order to explore possible ways of approaching this problem, I chose to use a deep neural network to model the dependencies between the features and the target value.

Due to the small dimensionality of the data (ranging from a 13-D to a 30-D space) I used a relatively simple but still powerful deep model, with the following architecture:

Figure 33: Deep neural network architecture



All intermediate connections between layers are regularized using dropout, which is an effective way to ensemble the network and reduce the intrinsic variance of the data [4].

The network was trained for different numbers of epochs based on the dataset (on some - e.g. Q2 - more training epochs were needed to reach an acceptable training and testing loss) using the mean squared error as loss and the Adam optimizer as training algorithm [2].

5 Experiments

A summary follows on the results obtained by the model when fitted on the different datasets.

Due to the small amount of samples available in the datasets, it was possible to evaluate the model performance using Leave-One-Out cross validation, and the results obtained are therefore the closest possible to real life performance on new samples.

However, since the main purpose of this model is to provide estimates at design time of a new system, other experiments were conducted to evaluate the model's interpolation (e.g. train on 16, 24, 40 nCores and test on 32 nCores) and generalization (e.g. train on 16, 24, 32 nCores and test on 40 nCores) capabilities.

To have a broader perspective of the performance obtained when testing interpolation and generalization, I computed the average predicted value and

measured some error metrics (RMSE, mean absolute error and mean relative error) against the average real value that the model should have produced. To show the capabilities of the model, these metrics were computed by grouping the predictions by the nCores and dataSize features.

Note that all the error metrics were computed on both the normalized values used by the model and the values in the original scale.

5.1 Dataset R1

5.1.1 Performance summary

RMSE scaled data	0.2546154702
RMSE original data	92330.2526413
MAE scaled data	0.1230911682
MAE original data	44636.0992133
MRE	0.0876223996

Figure 34: Average prediction of the model vs. average real value of the target feature, grouped by nCores and dataSize.

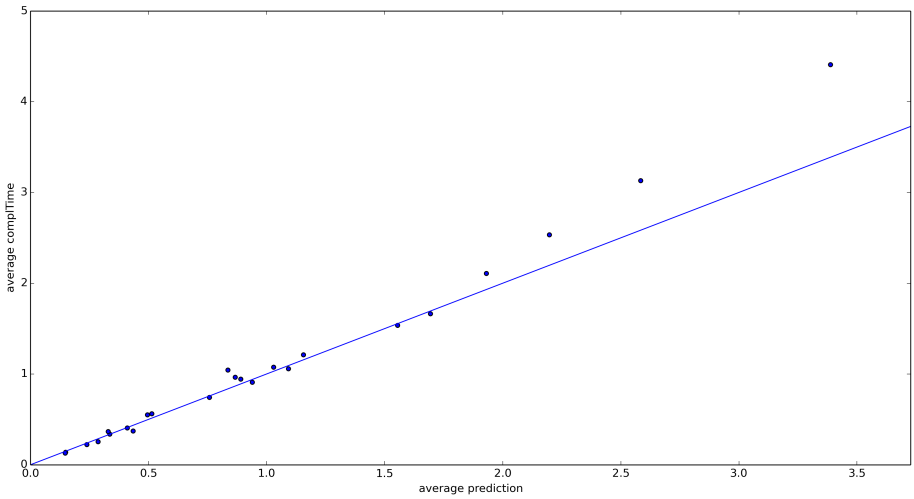
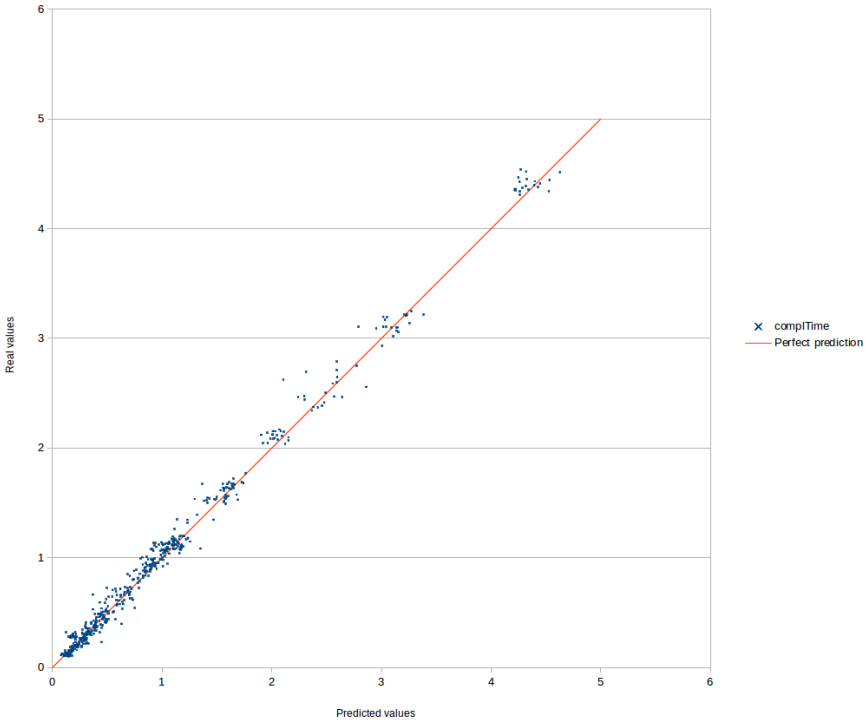


Figure 35: Predicted vs. real, LOO - Dataset R1

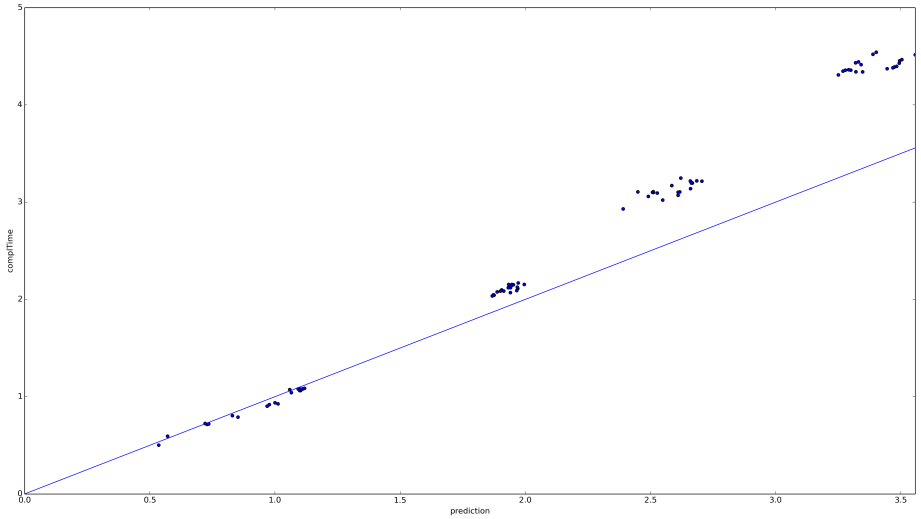


5.1.2 Testing details

Testing on 20 nCores

RMSE scaled data	0.5870311588
RMSE original data	212872.894452
MAE scaled data	0.4440190391
MAE original data	161012.975
MRE	0.1322468377

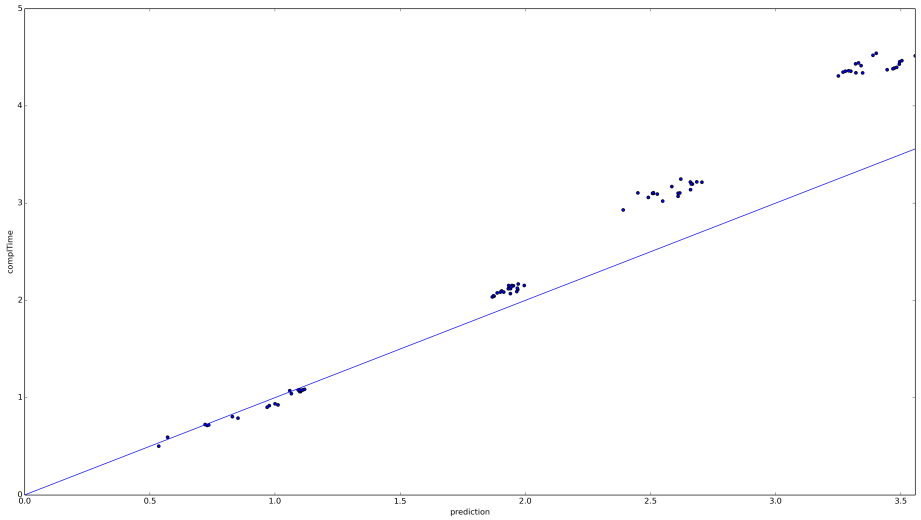
Figure 36: Predicted vs. real, test on 20 nCores - Dataset R1



Testing on 40 nCores

RMSE scaled data	0.1766855775
RMSE original data	64070.8771488
MAE scaled data	0.1065032986
MAE original data	38620.9375
MRE	0.0600707185

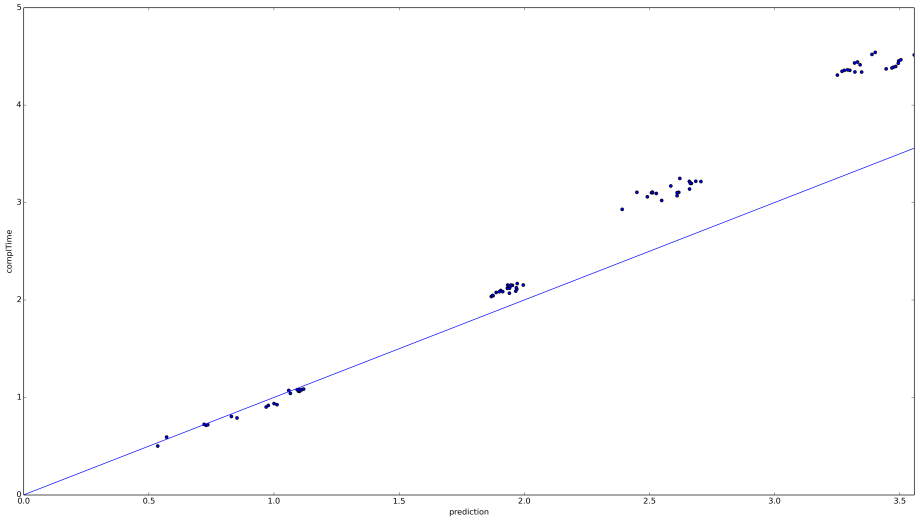
Figure 37: Predicted vs. real, test on 40 nCores - Dataset R1



Testing on 60 nCores

RMSE scaled data	0.1122916702
RMSE original data	40719.9136256
MAE scaled data	0.0744082369
MAE original data	26982.425
MRE	0.0889179625

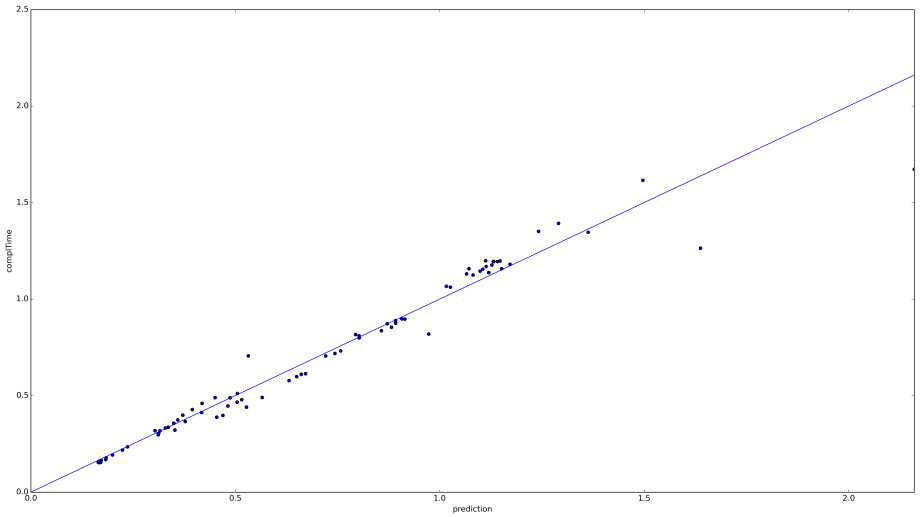
Figure 38: Predicted vs. real, test on 60 nCores - Dataset R1



Testing on 80 nCores

RMSE scaled data	0.0774265372
RMSE original data	28076.8622177
MAE scaled data	0.4440190391
MAE original data	161012.975
MRE	0.0646659282

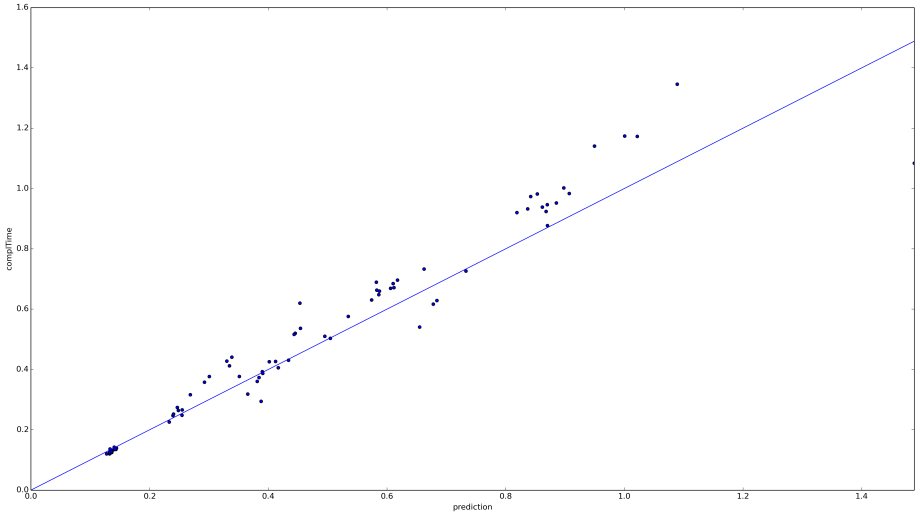
Figure 39: Predicted vs. real, test on 80 nCores - Dataset R1



Testing on 100 nCores

RMSE scaled data	0.0853656599
RMSE original data	30955.7546262
MAE scaled data	0.0558706521
MAE original data	20260.125
MRE	0.0968064287

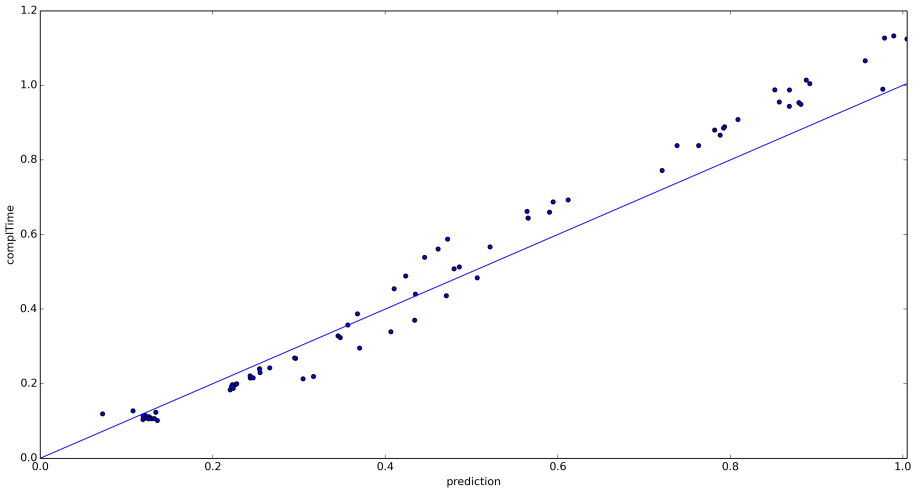
Figure 40: Predicted vs. real, test on 100 nCores - Dataset R1



Testing on 120 nCores

RMSE scaled data	0.0666493952
RMSE original data	24168.8170932
MAE scaled data	0.0540096364
MAE original data	19585.3093597
MRE	0.1327953594

Figure 41: Predicted vs. real, test on 120 nCores - Dataset R1



5.2 Dataset R2

5.2.1 Performance summary

RMSE scaled data	0.2799176044
RMSE original data	333434.317354
MAE scaled data	0.082094714
MAE original data	97790.2107906
MRE	0.1110719553

Figure 42: Average prediction of the model vs. average real value of the target feature, grouped by nCores and dataSize.

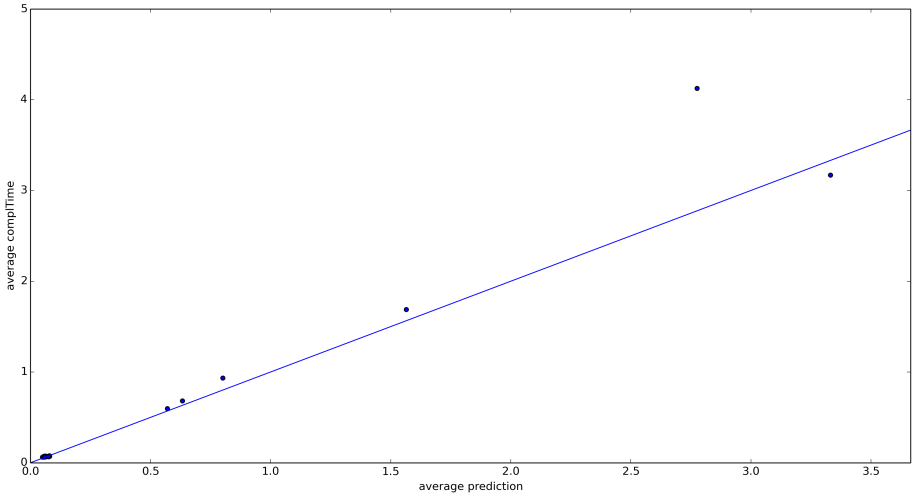
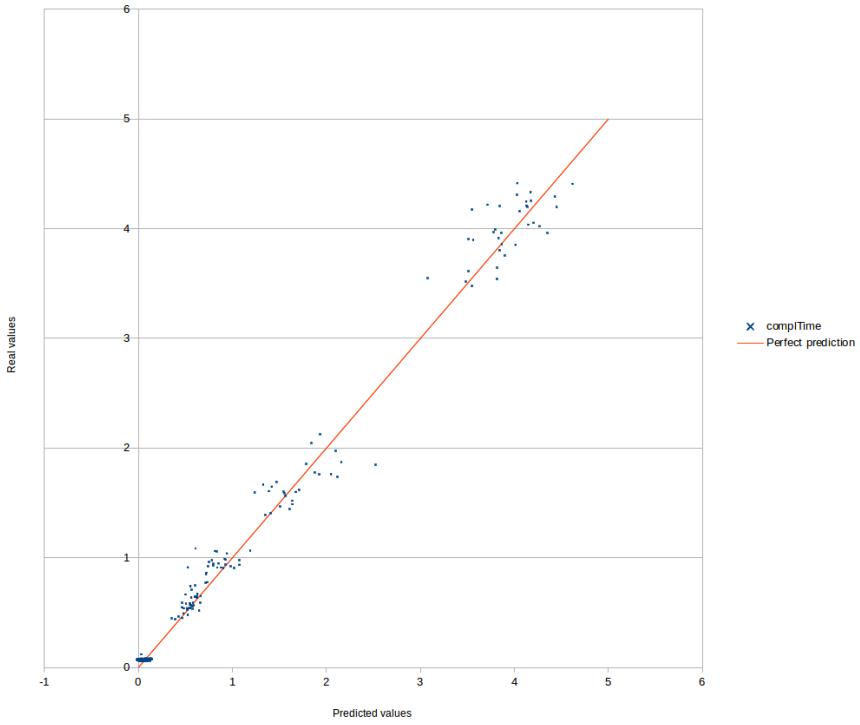


Figure 43: Predicted vs. real, LOO - Dataset R2

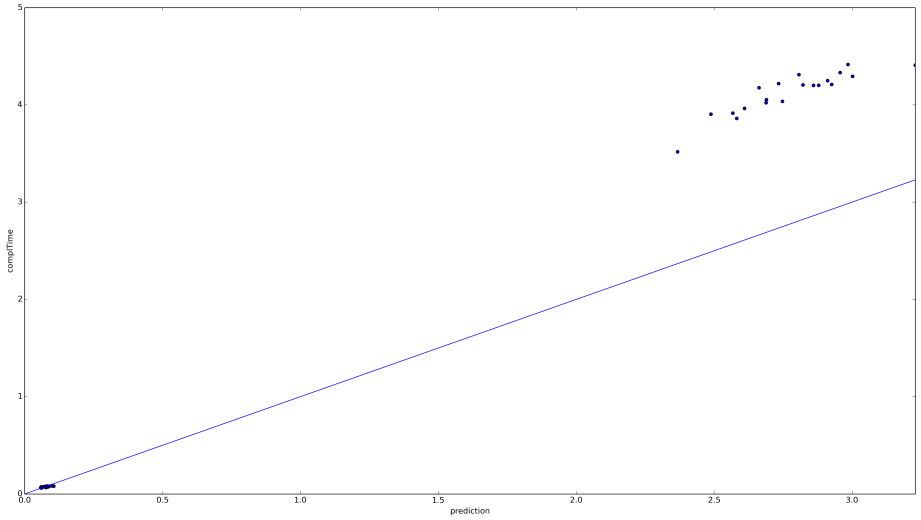


5.2.2 Testing details

Testing on 20 nCores

RMSE scaled data	0.6756742918
RMSE original data	804854.686066
MAE scaled data	0.341761886
MAE original data	407102.4875
MRE	0.1485109697

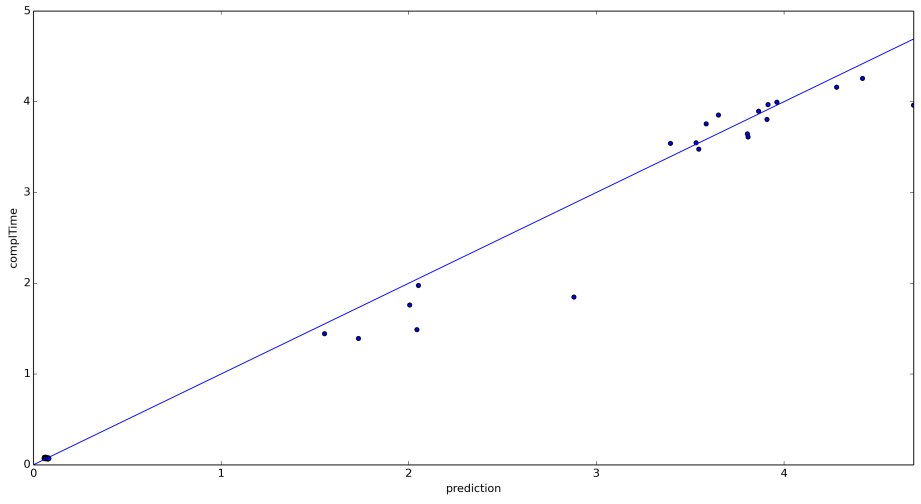
Figure 44: Predicted vs. real, test on 20 nCores - Dataset R2



Testing on 40 nCores

RMSE scaled data	0.170515129
RMSE original data	203115.460056
MAE scaled data	0.0633165293
MAE original data	75421.9
MRE	0.115979498

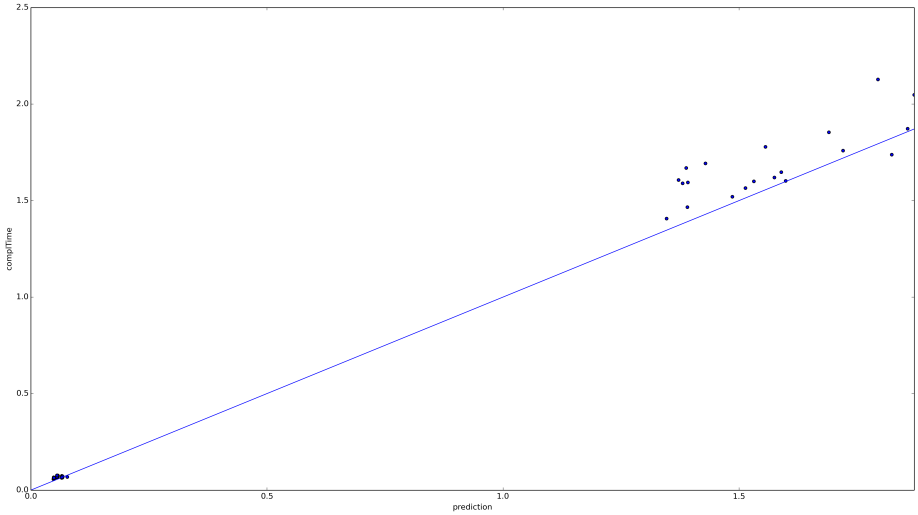
Figure 45: Predicted vs. real, test on 40 nCores - Dataset R2



Testing on 60 nCores

RMSE scaled data	0.0825694169
RMSE original data	98355.6743241
MAE scaled data	0.0398935427
MAE original data	47520.725
MRE	0.1265094506

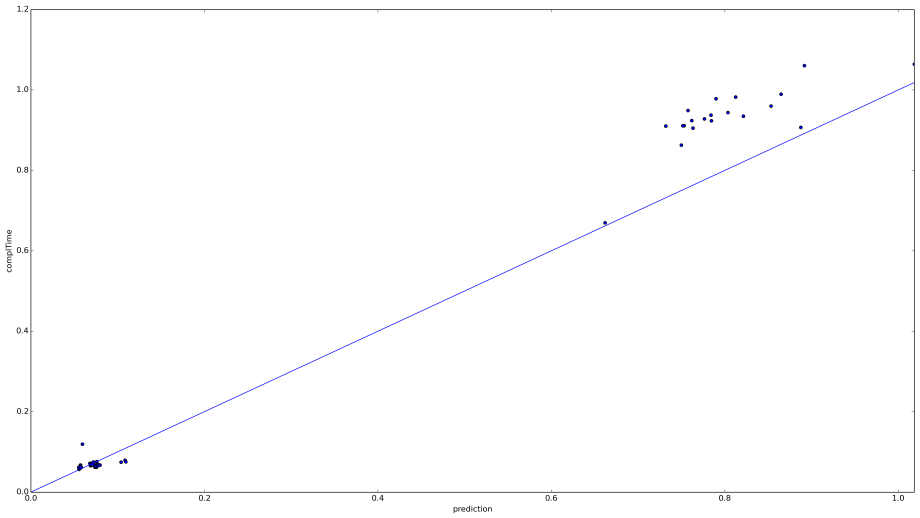
Figure 46: Predicted vs. real, test on 60 nCores - Dataset R2



Testing on 80 nCores

RMSE scaled data	0.0641477523
RMSE original data	76412.0020712
MAE scaled data	0.0325050767
MAE original data	38719.6666667
MRE	0.1143070846

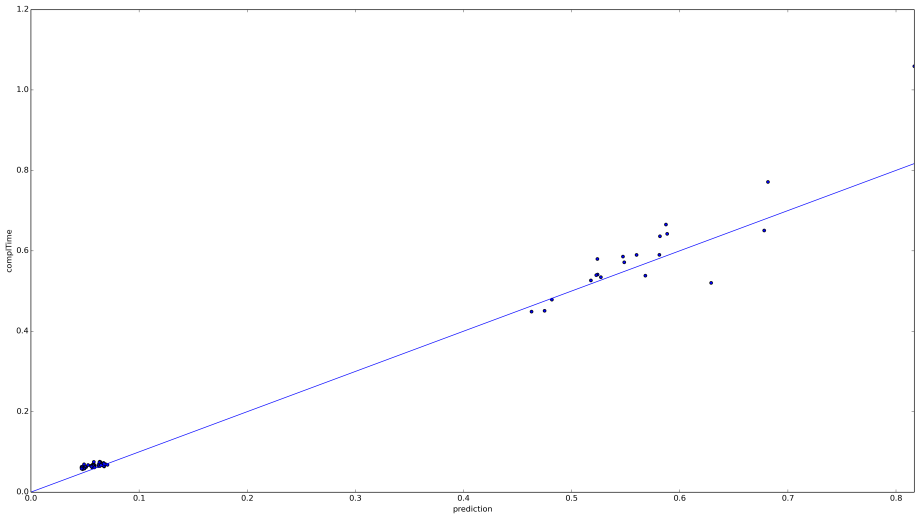
Figure 47: Predicted vs. real, test on 80 nCores - Dataset R2



Testing on 100 nCores

RMSE scaled data	0.0362101708
RMSE original data	43133.045293
MAE scaled data	0.0178335226
MAE original data	21243.075
MRE	0.1126907952

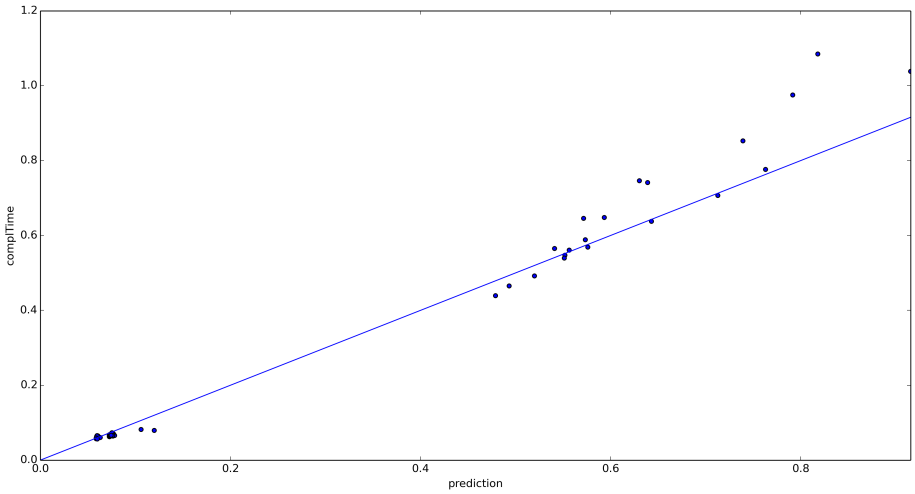
Figure 48: Predicted vs. real, test on 100 nCores - Dataset R2



Testing on 120 nCores

RMSE scaled data	0.0466531818
RMSE original data	55572.6659788
MAE scaled data	0.0200664848
MAE original data	23902.9125
MRE	0.0898904431

Figure 49: Predicted vs. real, test on 120 nCores - Dataset R2



5.3 Dataset R3

5.3.1 Performance summary

RMSE scaled data	0.2900537863
RMSE original data	258065.651414
MAE scaled data	0.1305585104
MAE original data	116160.053638
MRE	0.0793066637

Figure 50: Average prediction of the model vs. average real value of the target feature, grouped by nCores and dataSize.

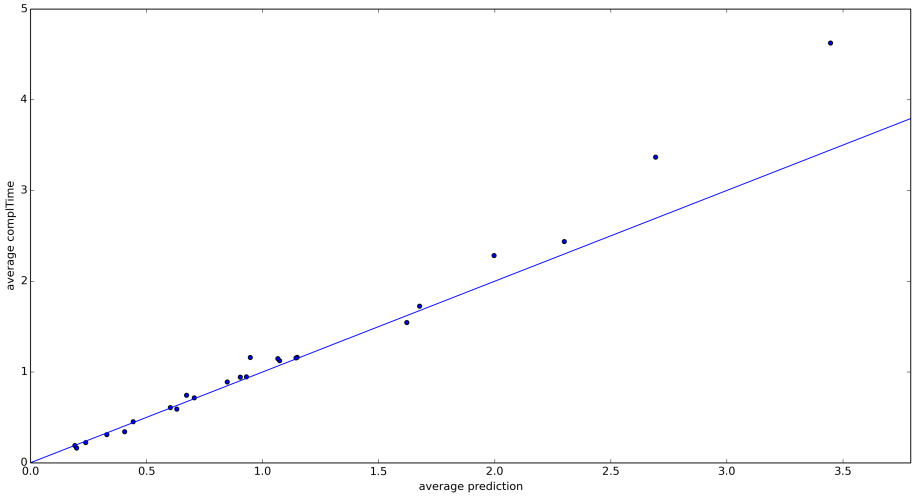
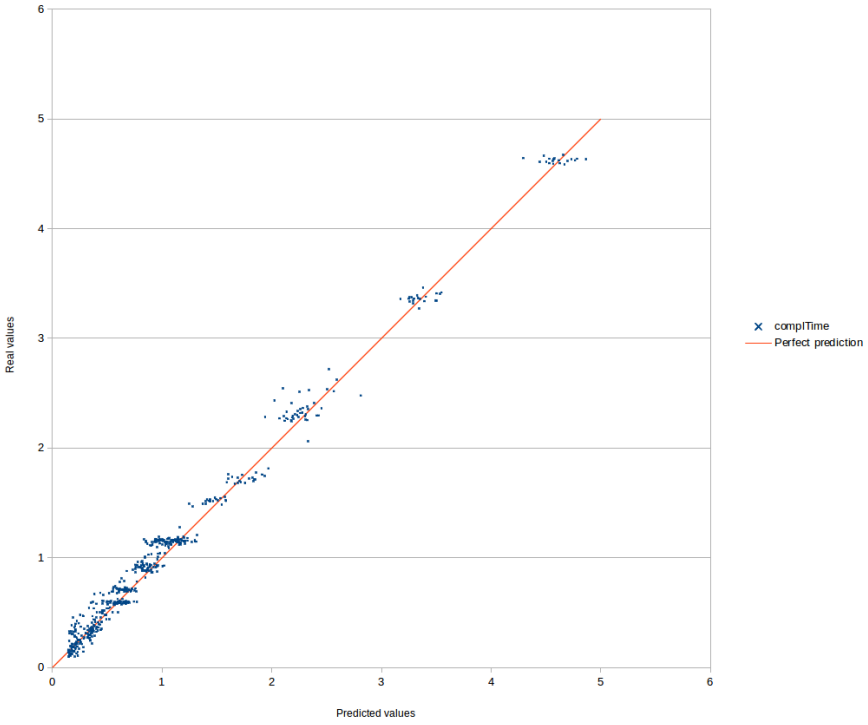


Figure 51: Predicted vs. real, LOO - Dataset R3

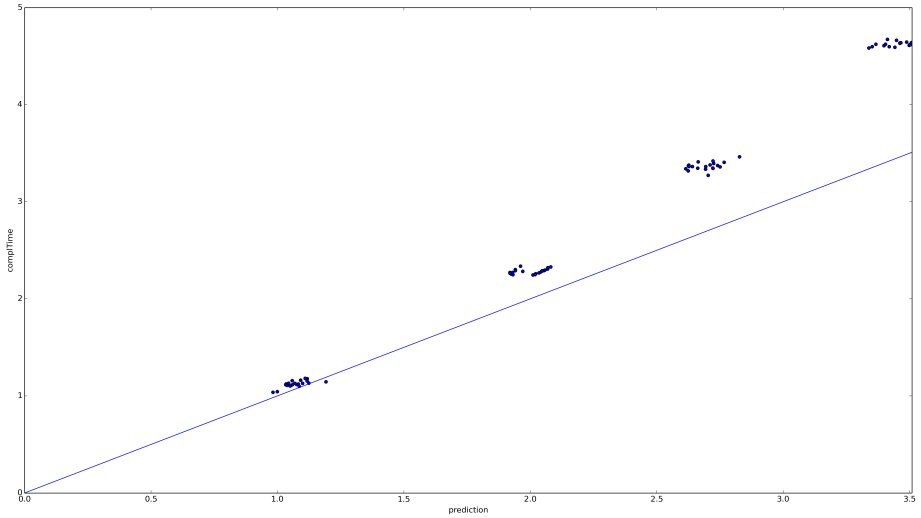


5.3.2 Testing details

Testing on 20 nCores

RMSE scaled data	0.686809335
RMSE original data	611065.618553
MAE scaled data	0.539808608
MAE original data	480276.632911
MRE	0.1559020898

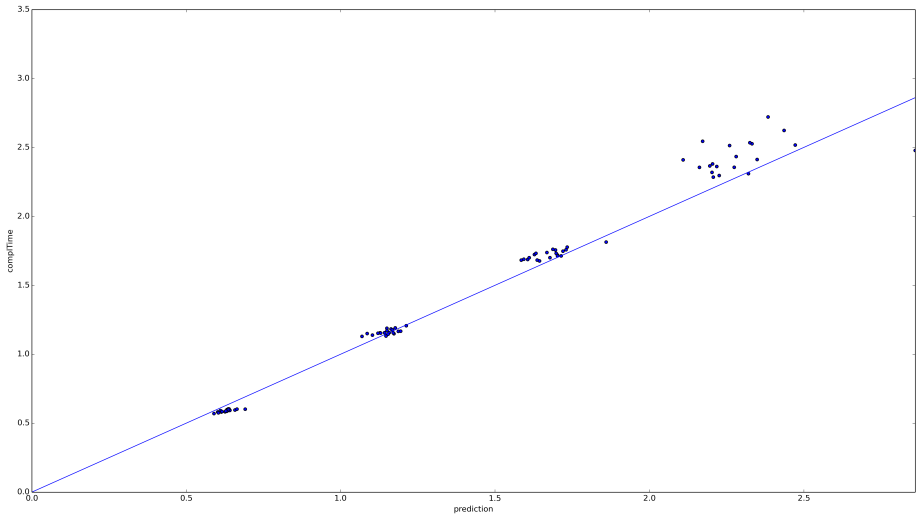
Figure 52: Predicted vs. real, test on 20 nCores - Dataset R3



Testing on 40 nCores

RMSE scaled data	0.110519463
RMSE original data	98331.0152372
MAE scaled data	0.0734633822
MAE original data	65361.6125
MRE	0.047607609

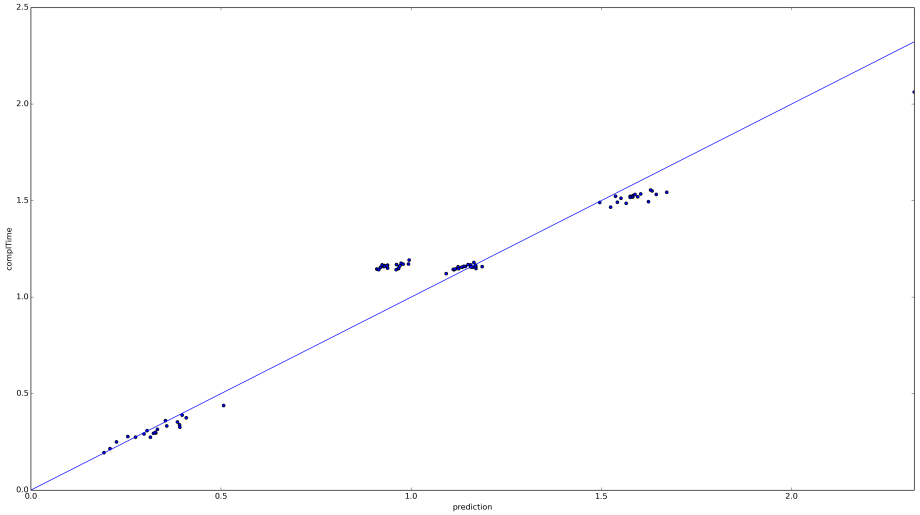
Figure 53: Predicted vs. real, test on 40 nCores - Dataset R3



Testing on 60 nCores

RMSE scaled data	0.1179560423
RMSE original data	104947.474287
MAE scaled data	0.0834416052
MAE original data	74239.3625
MRE	0.0817215754

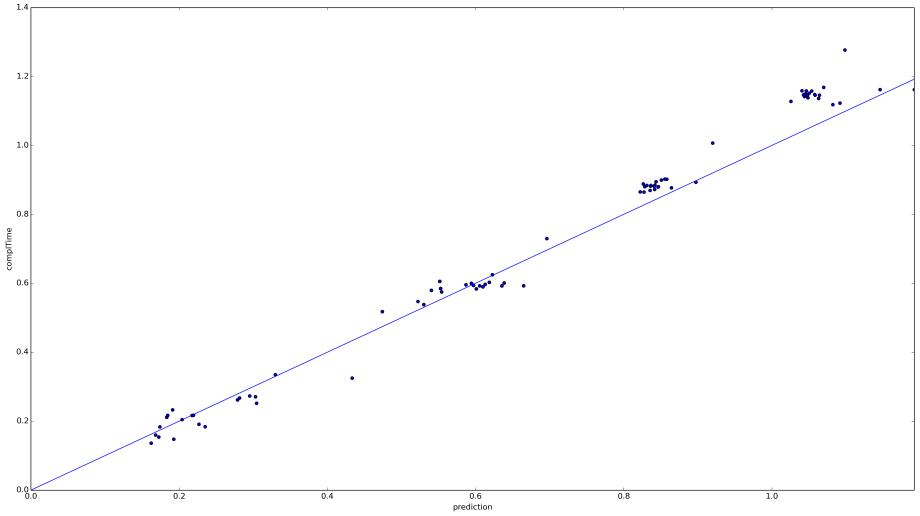
Figure 54: Predicted vs. real, test on 60 nCores - Dataset R3



Testing on 80 nCores

RMSE scaled data	0.0546403004
RMSE original data	48614.43369
MAE scaled data	0.0425680477
MAE original data	37873.5454545
MRE	0.0680262837

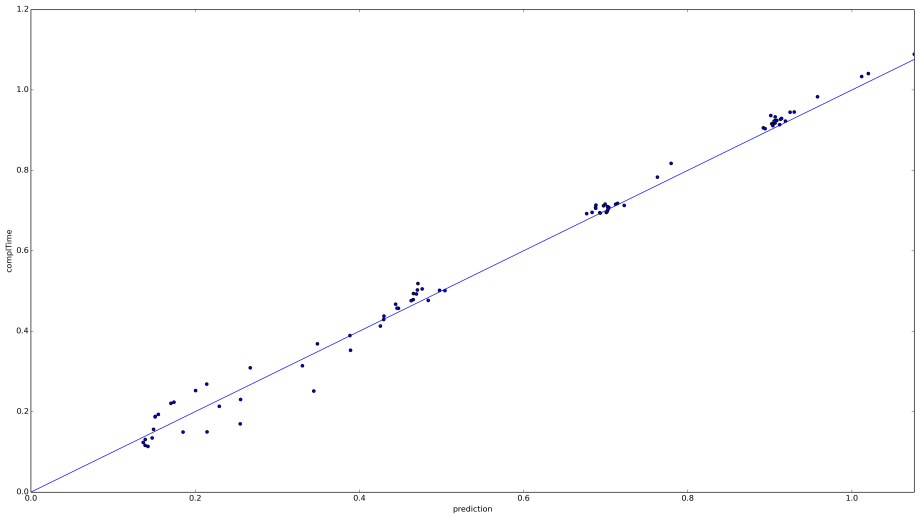
Figure 55: Predicted vs. real, test on 80 nCores - Dataset R3



Testing on 100 nCores

RMSE scaled data	0.0273479382
RMSE original data	24331.9414854
MAE scaled data	0.0205885234
MAE original data	18317.95
MRE	0.0682847594

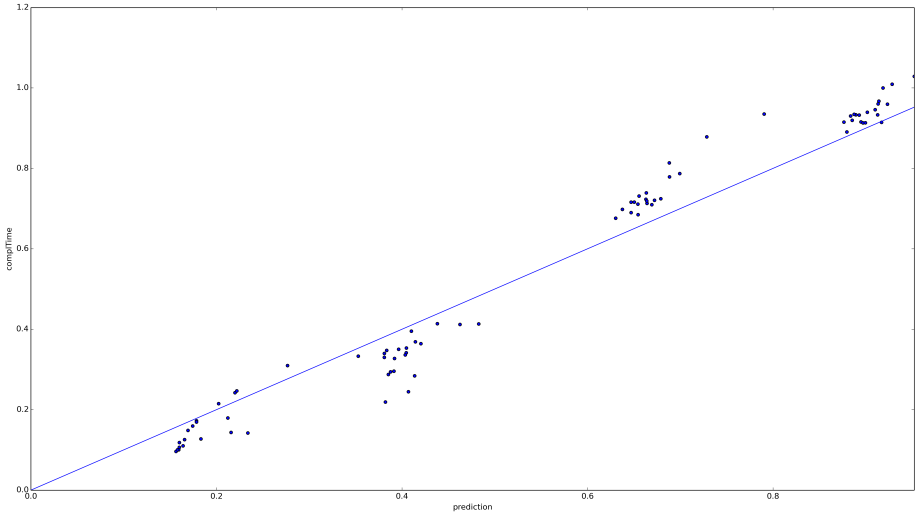
Figure 56: Predicted vs. real, test on 100 nCores - Dataset R3



Testing on 120 nCores

RMSE scaled data	0.0649128496
RMSE original data	57753.9859202
MAE scaled data	0.0551147558
MAE original data	49036.45
MRE	0.1677567326

Figure 57: Predicted vs. real, test on 120 nCores - Dataset R3



5.4 Dataset R4

5.4.1 Performance summary

RMSE scaled data	0.1151030673
RMSE original data	86152.6156117
MAE scaled data	0.0749137926
MAE original data	56071.6602564
MRE	0.0806297349

Figure 58: Average prediction of the model vs. average real value of the target feature, grouped by nCores and dataSize.

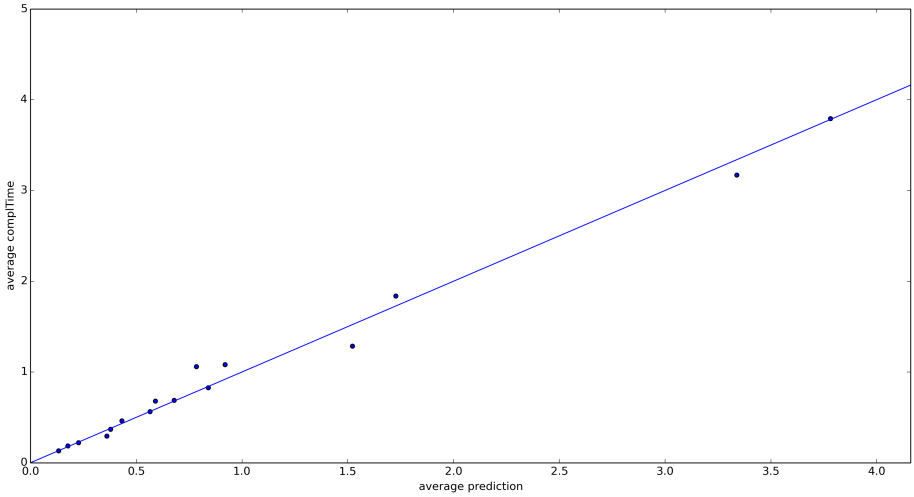
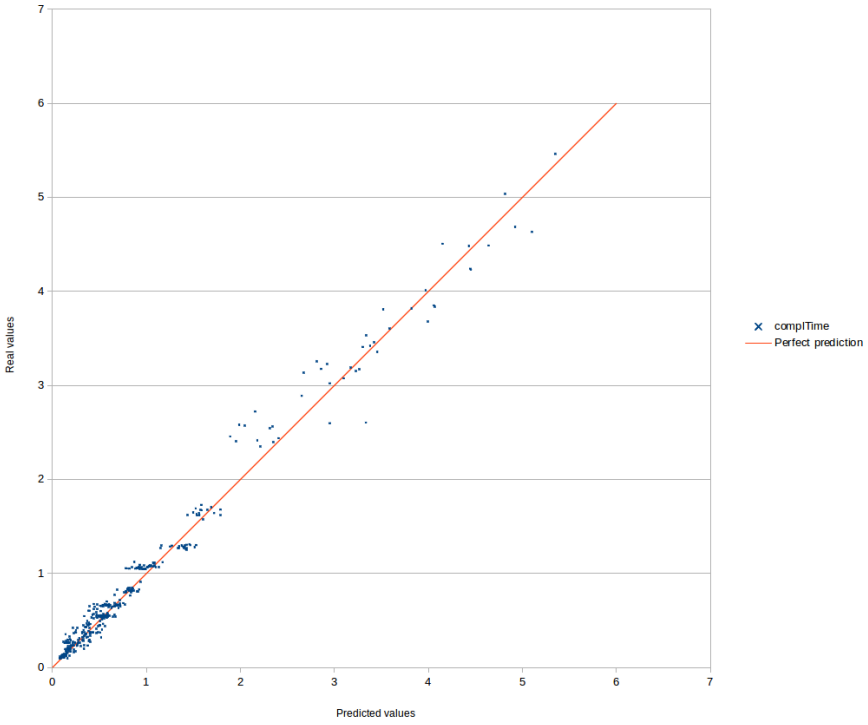


Figure 59: Predicted vs. real, LOO - Dataset R4

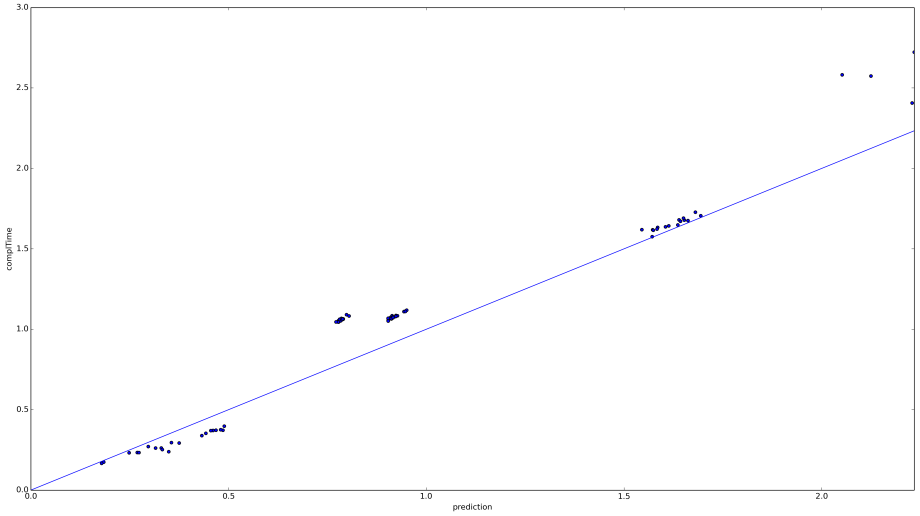


5.4.2 Testing details

Testing on 60 nCores

RMSE scaled data	0.1904972564
RMSE original data	142583.824089
MAE scaled data	0.1527446717
MAE original data	114326.6625
MRE	0.1693053198

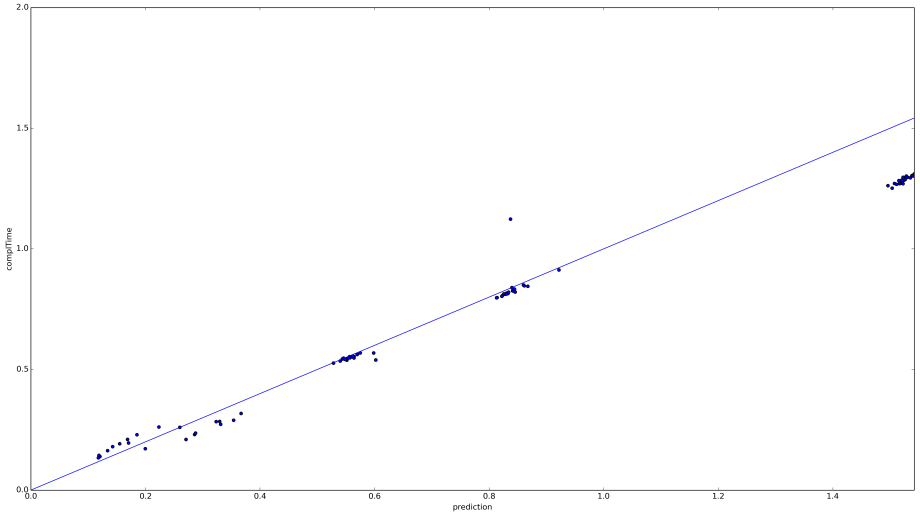
Figure 60: Predicted vs. real, test on 60 nCores - Dataset R4



Testing on 80 nCores

RMSE scaled data	0.1131943812
RMSE original data	84723.9955058
MAE scaled data	0.0658542923
MAE original data	49290.7676768
MRE	0.0862797486

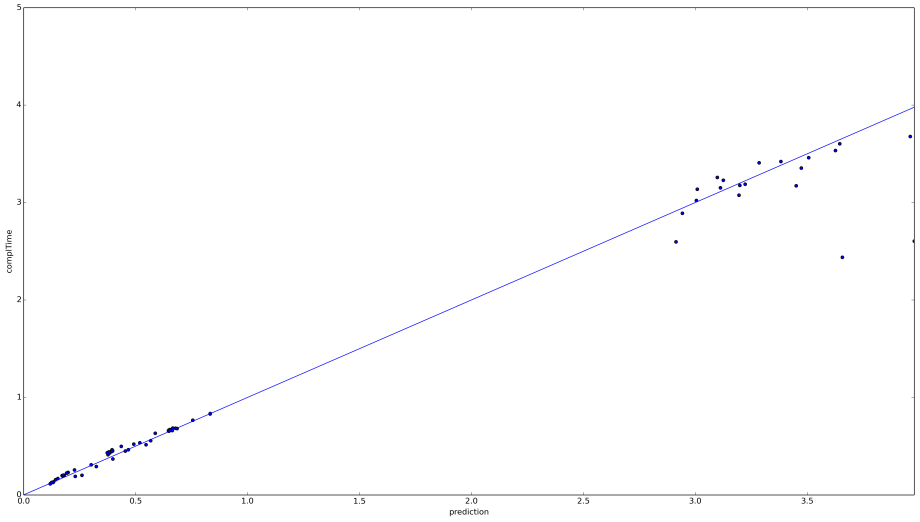
Figure 61: Predicted vs. real, test on 80 nCores - Dataset R4



Testing on 100 nCores

RMSE scaled data	0.2181235932
RMSE original data	163261.665346
MAE scaled data	0.0747000332
MAE original data	55911.6625
MRE	0.0702347057

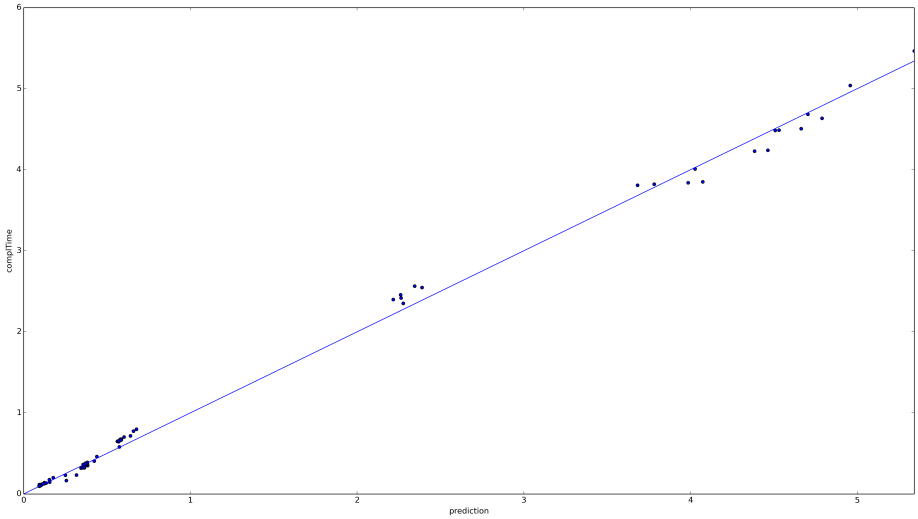
Figure 62: Predicted vs. real, test on 100 nCores - Dataset R4



Testing on 120 nCores

RMSE scaled data	0.0860030713
RMSE original data	64371.7894512
MAE scaled data	0.0614004809
MAE original data	45957.175
MRE	0.0790341969

Figure 63: Predicted vs. real, test on 120 nCores - Dataset R4



5.5 Dataset R5

5.5.1 Performance summary

RMSE scaled data	0.1272236417
RMSE original data	984.389468504
MAE scaled data	0.1091952774
MAE original data	844.887980769
MRE	0.0286438644

Figure 64: Average prediction of the model vs. average real value of the target feature, grouped by nCores and dataSize.

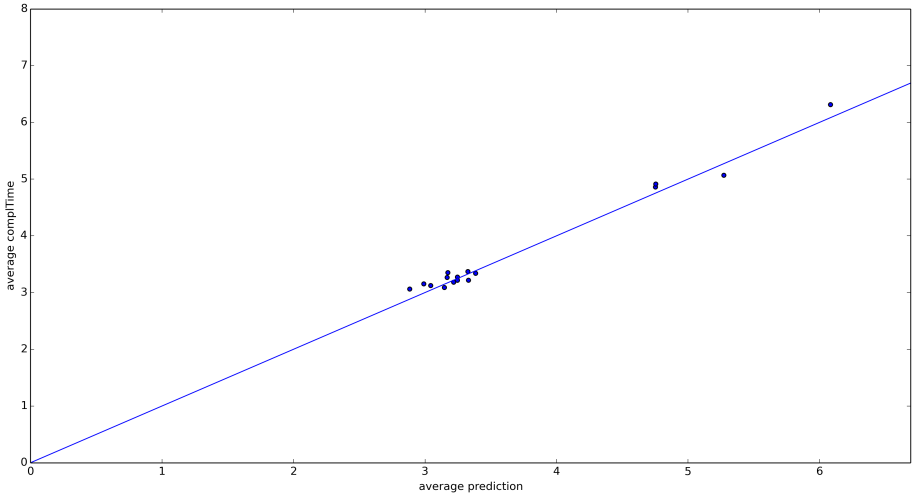
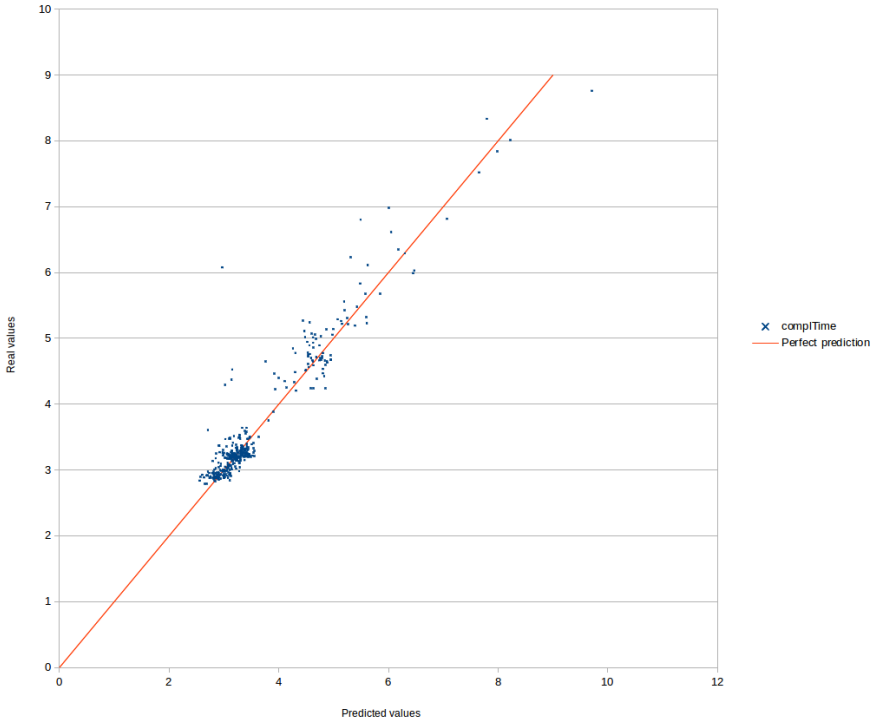


Figure 65: Predicted vs. real, LOO - Dataset R5

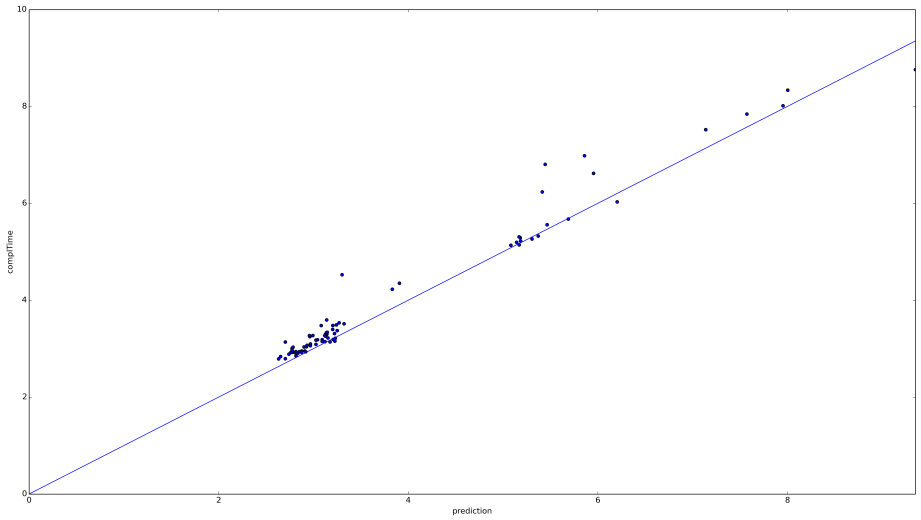


5.5.2 Testing details

Testing on 60 nCores

RMSE scaled data	0.328894004
RMSE original data	2544.77798148
MAE scaled data	0.2123823647
MAE original data	1643.325
MRE	0.051170406

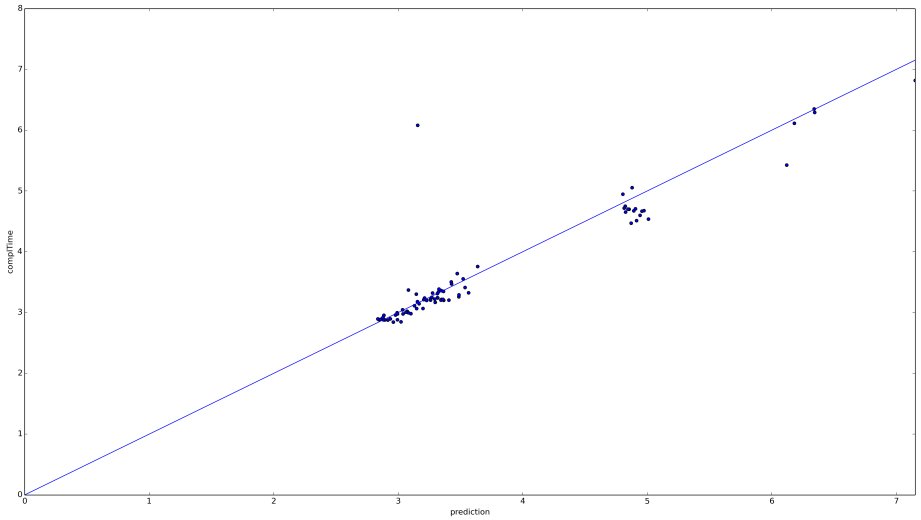
Figure 66: Predicted vs. real, test on 60 nCores - Dataset R5



Testing on 80 nCores

RMSE scaled data	0.332237912
RMSE original data	2570.6542185
MAE scaled data	0.1335914318
MAE original data	1033.63636364
MRE	0.0323345181

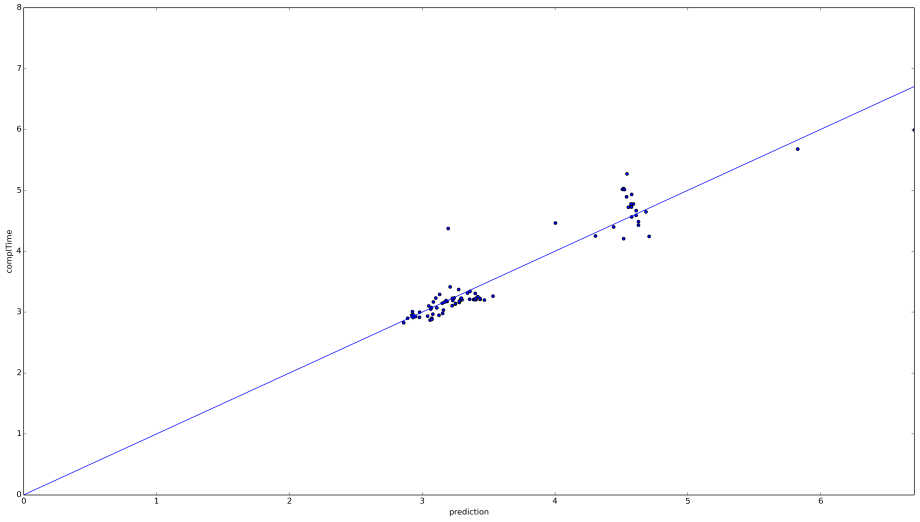
Figure 67: Predicted vs. real, test on 80 nCores - Dataset R5



Testing on 100 nCores

RMSE scaled data	0.2520468928
RMSE original data	1950.15510281
MAE scaled data	0.1649368251
MAE original data	1276.2
MRE	0.0419571908

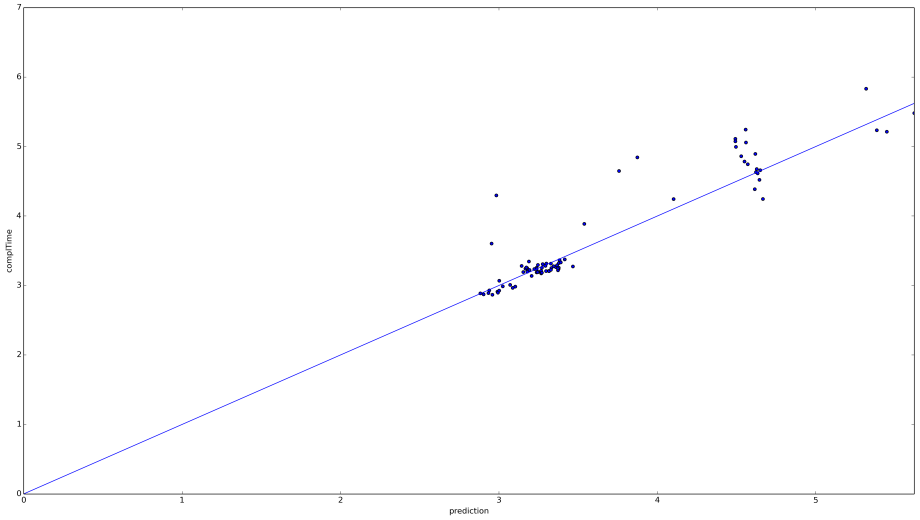
Figure 68: Predicted vs. real, test on 100 nCores - Dataset R5



Testing on 120 nCores

RMSE scaled data	0.2946807325
RMSE original data	2279.9836211
MAE scaled data	0.1729817721
MAE original data	1338.3875
MRE	0.0416949136

Figure 69: Predicted vs. real, test on 120 nCores - Dataset R5



5.6 Dataset Q2

5.6.1 Performance summary

RMSE scaled data	0.2549837666
RMSE original data	12352.5686963
MAE scaled data	0.2141470056
MAE original data	10374.2634259
MRE	0.0406312385

Figure 70: Average prediction of the model vs. average real value of the target feature, grouped by nCores and dataSize.

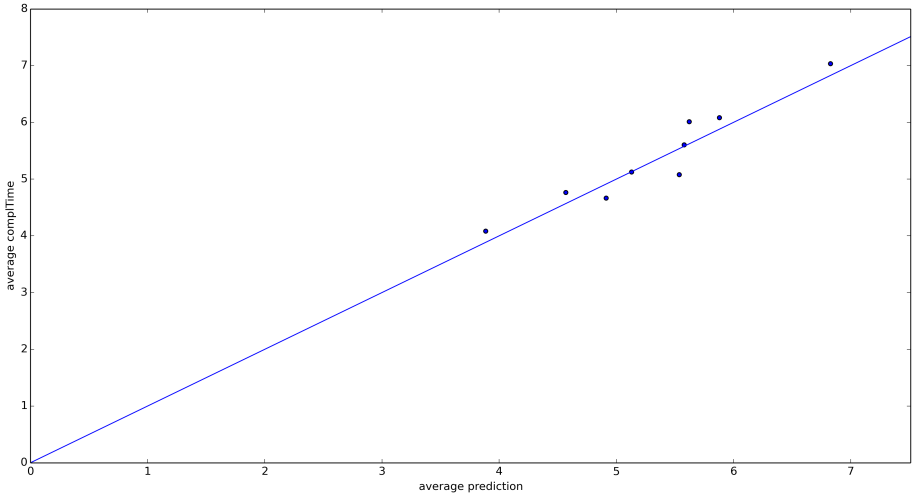
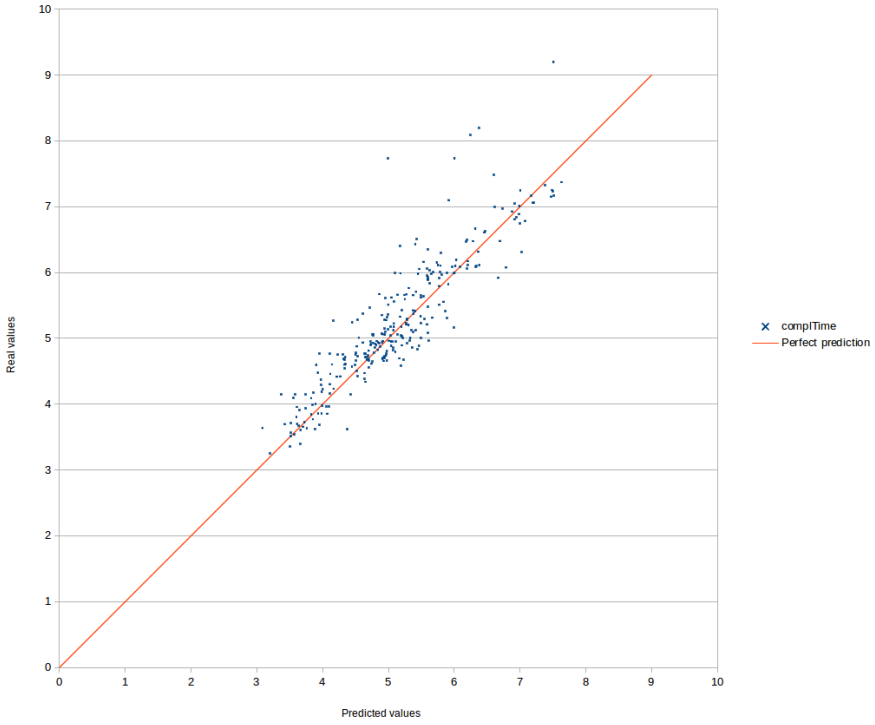


Figure 71: Predicted vs. real, LOO - Dataset Q2

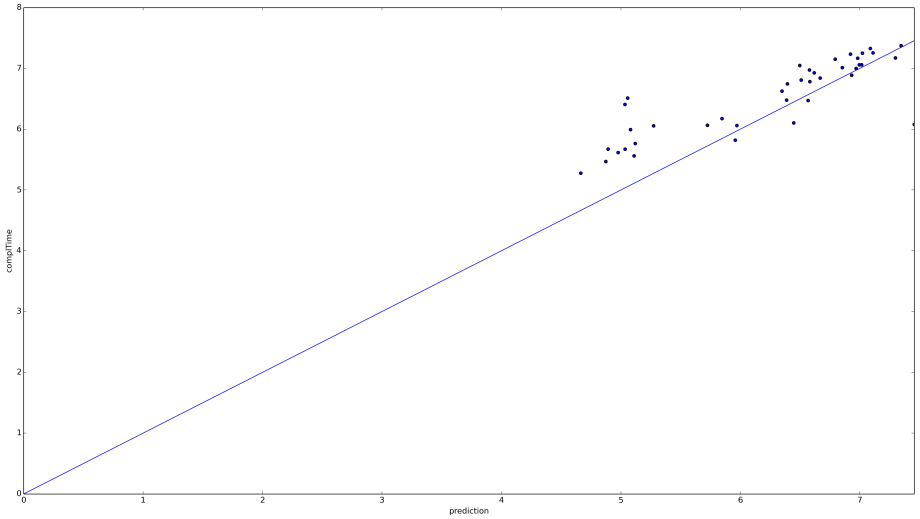


5.6.2 Testing details

Testing on 16 nCores

RMSE scaled data	0.5436355311
RMSE original data	26336.1861746
MAE scaled data	0.4041283907
MAE original data	19577.825
MRE	0.0649931381

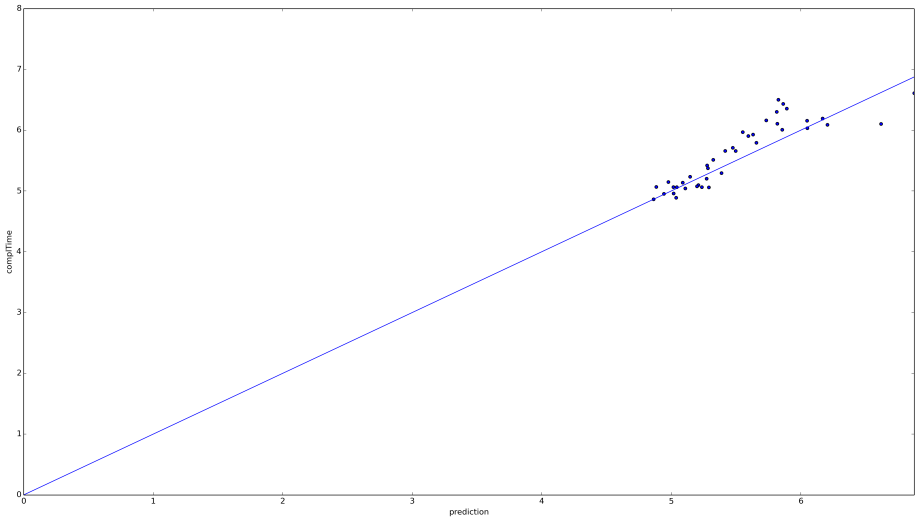
Figure 72: Predicted vs. real, test on 16 nCores - Dataset Q2



Testing on 24 nCores

RMSE scaled data	0.2590079795
RMSE original data	12547.5006585
MAE scaled data	0.1990050231
MAE original data	9640.725
MRE	0.0339976637

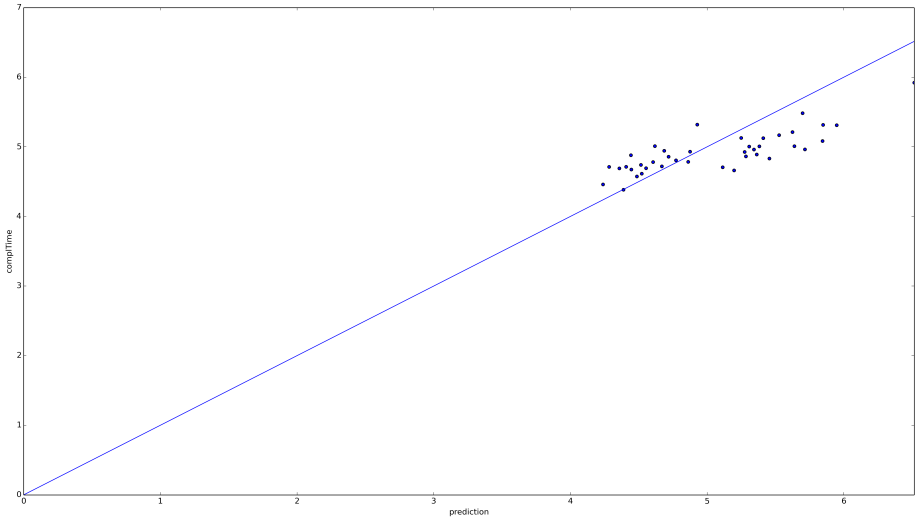
Figure 73: Predicted vs. real, test on 24 nCores - Dataset Q2



Testing on 32 nCores

RMSE scaled data	0.3867288435
RMSE original data	18734.8797487
MAE scaled data	0.3312977819
MAE original data	16049.55
MRE	0.0665019798

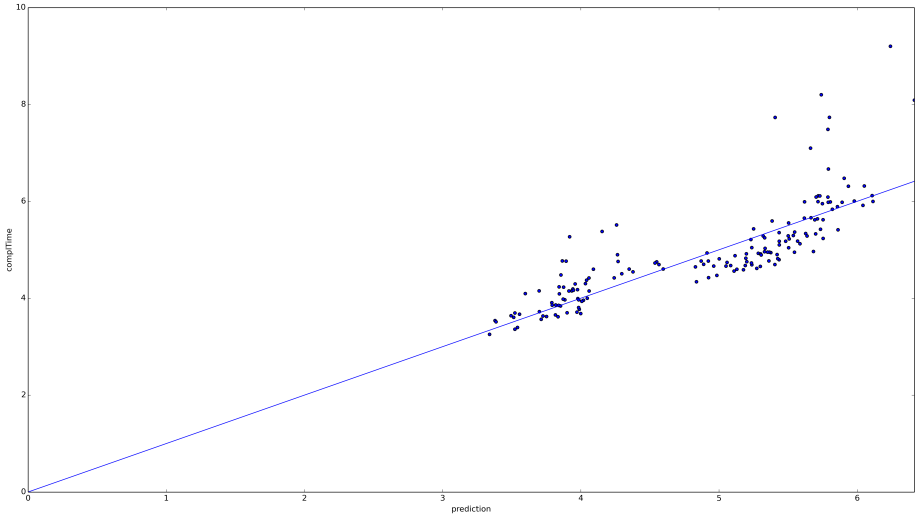
Figure 74: Predicted vs. real, test on 32 nCores - Dataset Q2



Testing on 40 nCores

RMSE scaled data	0.5842215785
RMSE original data	28302.2856689
MAE scaled data	0.3800912654
MAE original data	18413.30625
MRE	0.0715549642

Figure 75: Predicted vs. real, test on 40 nCores - Dataset Q2



5.7 Dataset Q3

5.7.1 Performance summary

RMSE scaled data	0.6870406969
RMSE original data	21582.4502322
MAE scaled data	0.5708905389
MAE original data	17933.7625
MRE	0.0834076827

Figure 76: Average prediction of the model vs. average real value of the target feature, grouped by nCores and dataSize.

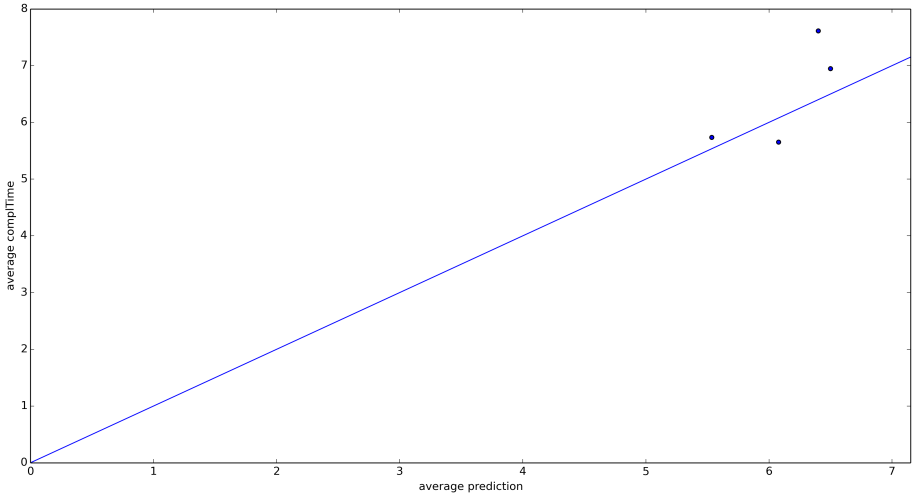
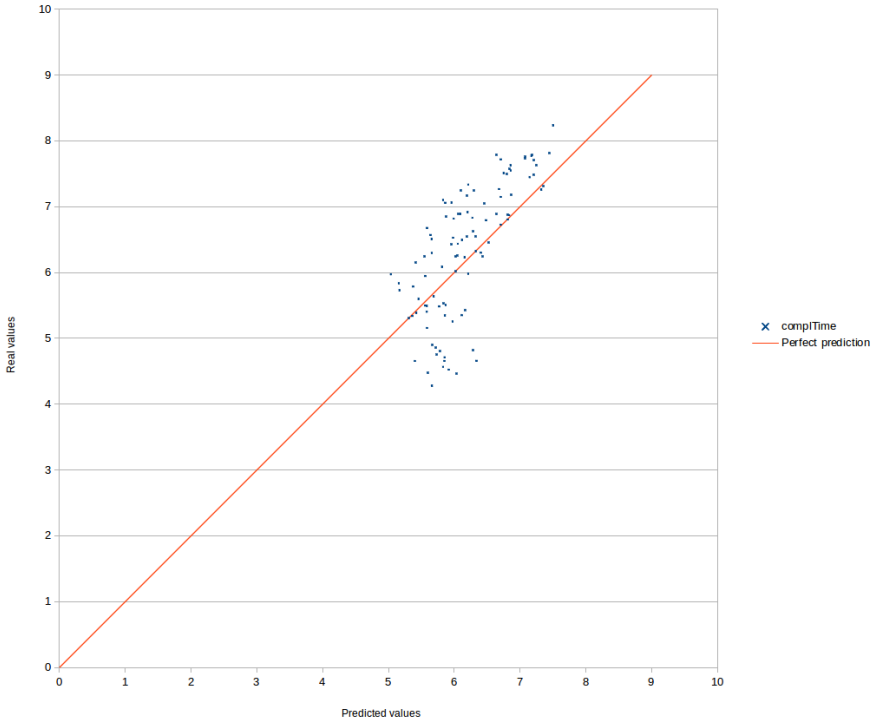


Figure 77: Predicted vs. real, LOO - Dataset Q3

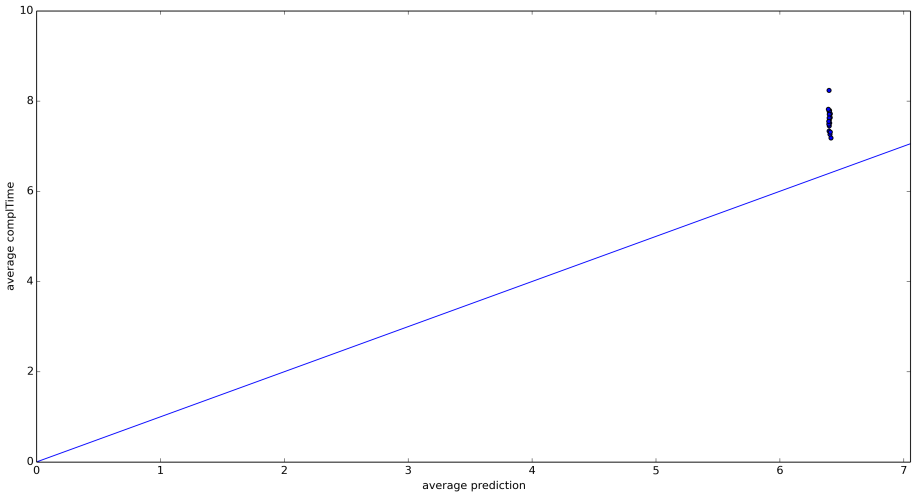


5.7.2 Testing details

Testing on 16 nCores

RMSE scaled data	1.2343232459
RMSE original data	38774.5714335
MAE scaled data	1.2111826733
MAE original data	38047.65
MRE	0.1583077962

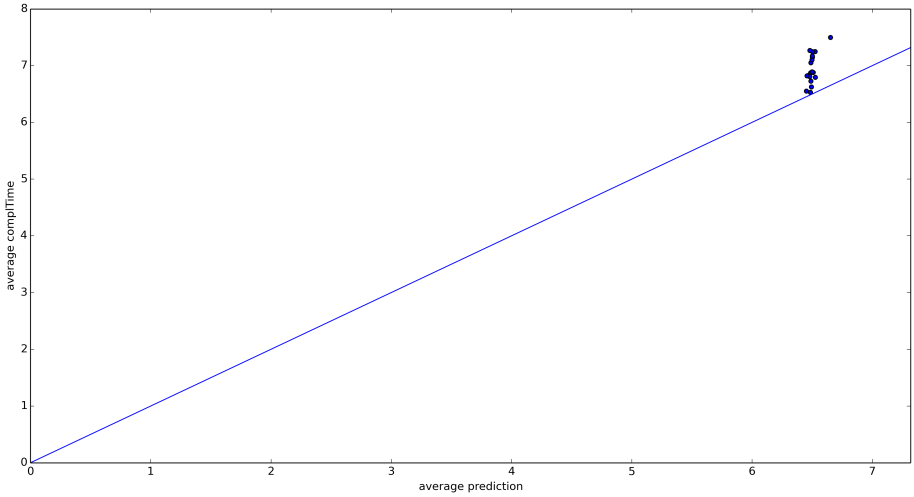
Figure 78: Predicted vs. real, test on 16 nCores - Dataset Q3



Testing on 24 nCores

RMSE scaled data	0.5027544532
RMSE original data	15793.3623716
MAE scaled data	0.446449041
MAE original data	14024.6
MRE	0.0631559665

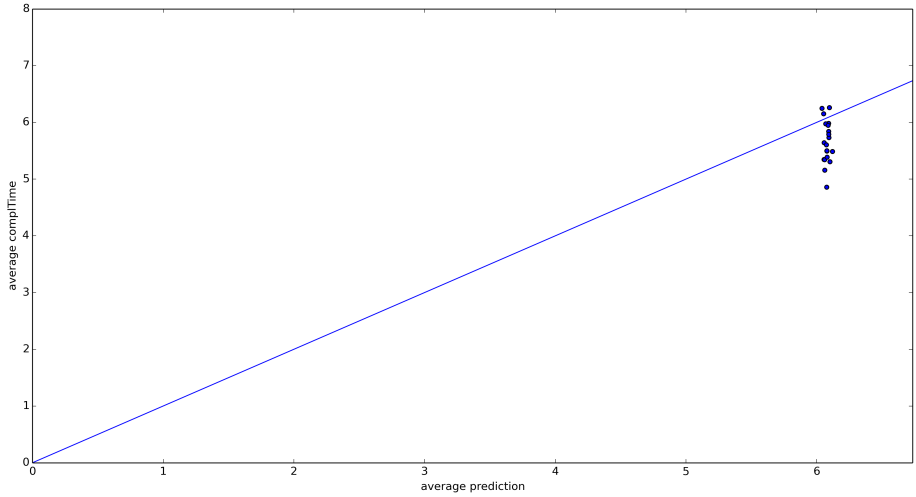
Figure 79: Predicted vs. real, test on 24 nCores - Dataset Q3



Testing on 32 nCores

RMSE scaled data	0.5627229404
RMSE original data	17677.1943588
MAE scaled data	0.4732774228
MAE original data	14867.4
MRE	0.0875542749

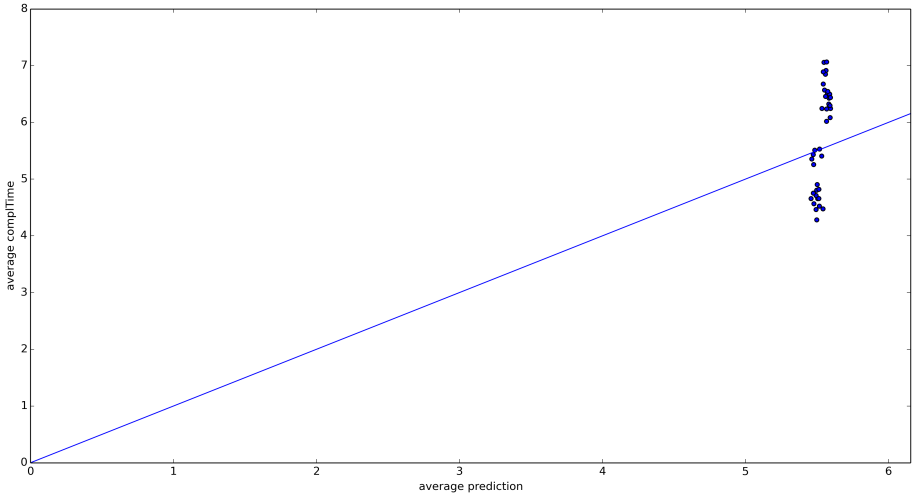
Figure 80: Predicted vs. real, test on 32 nCores - Dataset Q3



Testing on 40 nCores

RMSE scaled data	0.8732890966
RMSE original data	27433.2830518
MAE scaled data	0.7856081669
MAE original data	24678.9
MRE	0.1381667548

Figure 81: Predicted vs. real, test on 40 nCores - Dataset Q3



5.8 Dataset Q4

5.8.1 Performance summary

RMSE scaled data	0.5019232326
RMSE original data	33054.3019827
MAE scaled data	0.3549287533
MAE original data	23373.9625
MRE	0.0833578091

Figure 82: Average prediction of the model vs. average real value of the target feature, grouped by nCores and dataSize.

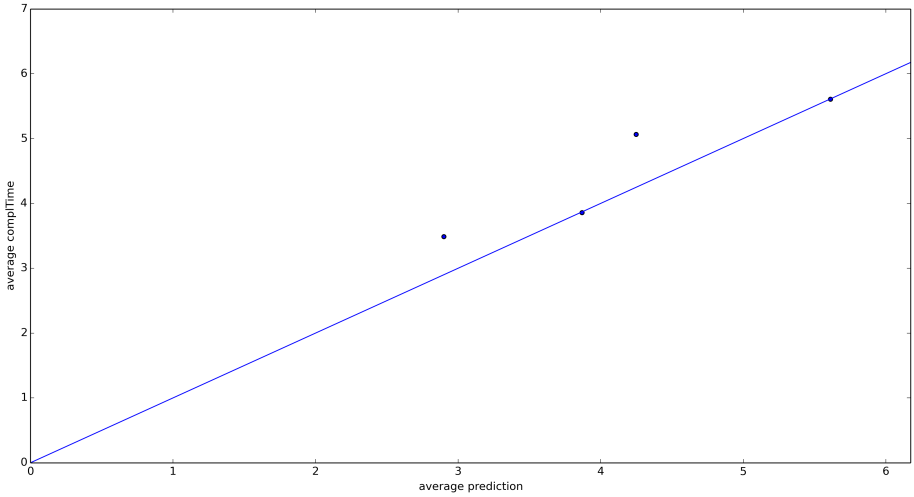
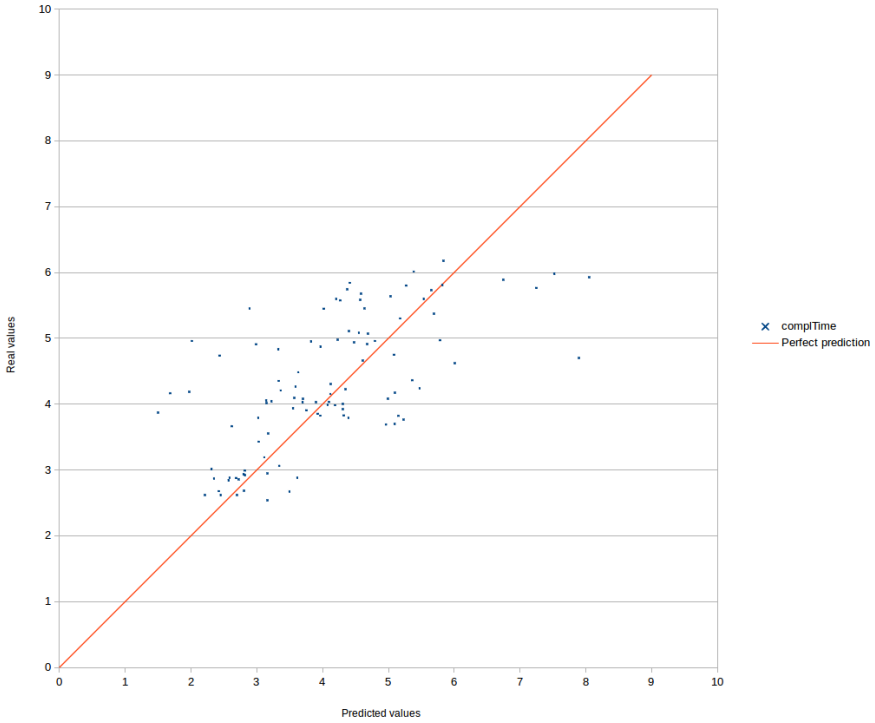


Figure 83: Predicted vs. real, LOO - Dataset Q4

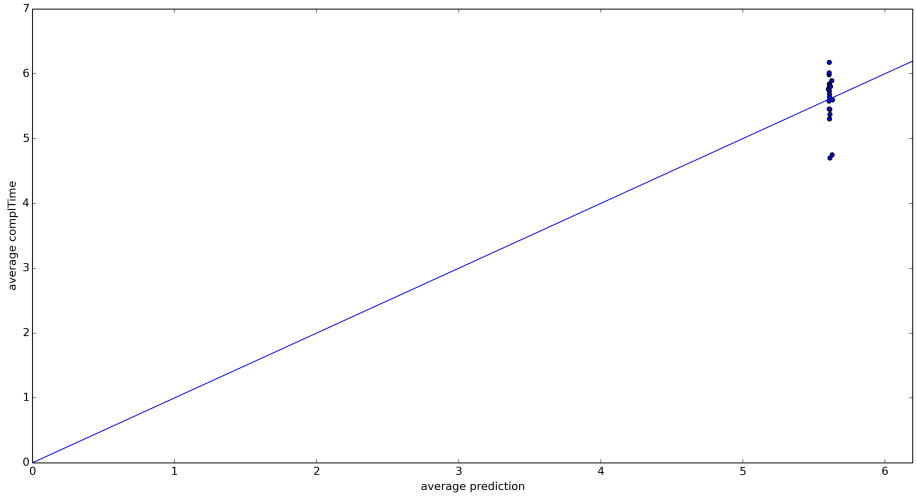


5.8.2 Testing details

Testing on 16 nCores

RMSE scaled data	0.3667002424
RMSE original data	24149.1243444
MAE scaled data	0.2685778478
MAE original data	17687.3
MRE	0.049909849

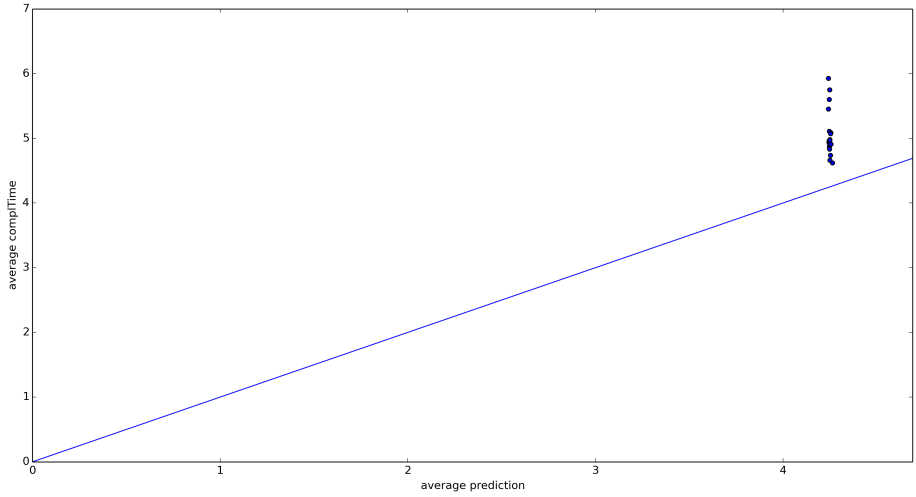
Figure 84: Predicted vs. real, test on 16 nCores - Dataset Q4



Testing on 24 nCores

RMSE scaled data	0.8839167981
RMSE original data	58210.5692972
MAE scaled data	0.8144815931
MAE original data	53637.9
MRE	0.1572670982

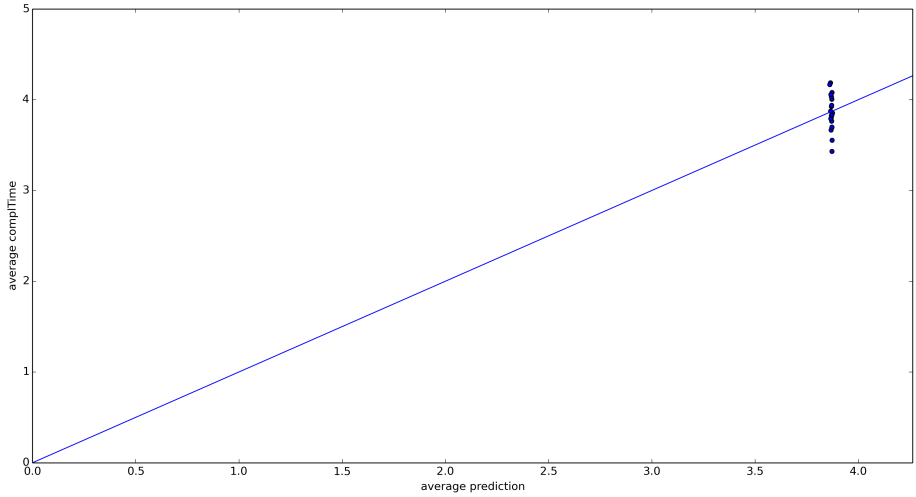
Figure 85: Predicted vs. real, test on 24 nCores - Dataset Q4



Testing on 32 nCores

RMSE scaled data	0.194390548
RMSE original data	12801.6984947
MAE scaled data	0.1569757408
MAE original data	10337.75
MRE	0.0411203343

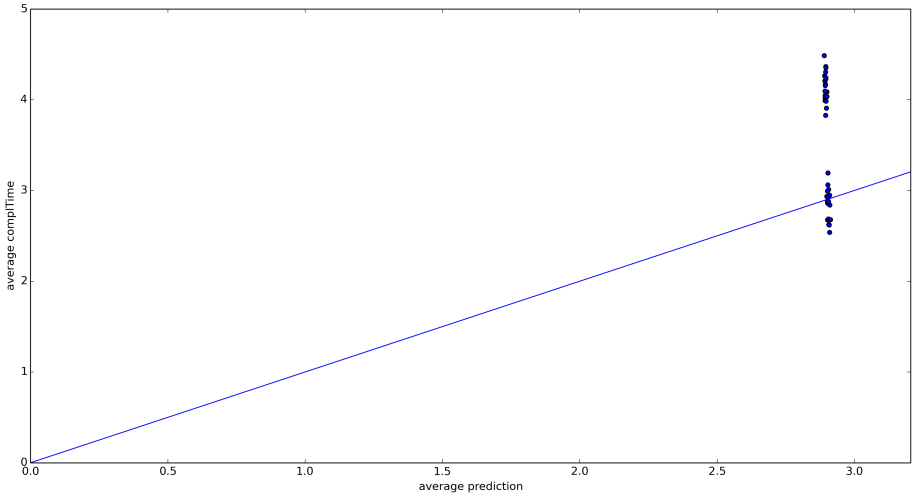
Figure 86: Predicted vs. real, test on 32 nCores - Dataset Q4



Testing on 40 nCores

RMSE scaled data	0.8967442267
RMSE original data	59055.4232349
MAE scaled data	0.6936369471
MAE original data	45679.7
MRE	0.1758408484

Figure 87: Predicted vs. real, test on 40 nCores - Dataset Q4



6 Conclusions

I showed that the model was able to learn the data in a satisfactory way on most of the datasets, with the biggest problems emerging on the Q datasets and in general when extrapolating on new data outside of the training domain. However, given that neural networks are not known for having strong generalization capabilities outside of the training domain, the obtained results exceed expectations.

I was able to obtain an average mean relative error around 10% on all datasets, with some datasets reaching MREs below 5%.

Further extensions of this work may imply the use of memory based methods (e.g. KNN) in order to tackle the fragmentation of the clusters that emerge when projecting the feature space in the nCores - complTime space.

List of Tables

List of Figures

References

[1] P. Ferretti A. Battistello. Machine learning techniques to model data intensive application performance. Master's thesis, Politecnico di Milano, 2015.

[2] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2014.

[3] Jason Venner. *Pro Hadoop*. Apress, Berkeley, CA, 2009.

[4] Stefan Wager, Sida Wang, and Percy S Liang. Dropout training as adaptive regularization. In C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 26*, pages 351–359. Curran Associates, Inc., 2013.