

Contents

1	Introduction	4
2	Test and evaluation methodology	4
3	Cajamar: test and evaluation	5
3.1	Use case requirements	5
3.2	Model and data characteristics	5
3.3	Predictive performance: test and evaluation	5
3.3.1	Application scenario 1	5
3.3.2	Application scenario 2	5
3.4	Run-time performance: test and evaluation	5
3.4.1	Application scenario 1	5
3.4.2	Application scenario 2	5
4	Daimler: test and evaluation	5
4.1	Use case requirements	5
4.2	Model and data characteristics	6
4.3	Predictive performance: test and evaluation	6
4.3.1	Application scenario 1	6
4.3.2	Application scenario 2	6
4.4	Run-time performance: test and evaluation	6
4.4.1	Application scenario 1	6
4.4.2	Application scenario 2	6
5	Verdande: test and evaluation	6
5.1	Use case requirements	6
5.2	Model and data characteristics	6
5.3	Predictive performance: test and evaluation	7
5.3.1	Application scenario 1	7
5.3.2	Application scenario 2	7

5.4	Run-time performance: test and evaluation	7
5.4.1	Application scenario 1	7
5.4.2	Application scenario 2	7
6	Conclusion	7

Document history

Version	Date	Author (Unit)	Description
v0.3			The test and evaluation framework discussed and established
v0.6			Initial draft finished and reviewed by the PSRG
v1.0			Final version of document

1 Introduction

Even though the number of algorithms designed for learning on streaming data is increasing, there is still not a unified and well accepted way for evaluating them. On one side, there are statistical challenges related to the fact that data are generated from underlying distributions that are time dependent and also that the algorithms themselves are time dependent. Moreover, the fact that the data come from an open-ended data stream, conceptually infinitely long, imposes practical challenges related to restrictions on cpu-time and memory allocation.

In this paper we will establish formal procedures for testing and evaluating the developed models and algorithms. This includes specification what metrics are relevant to use to quantify the ability of the AMIDST system, such as relevant formalization of loss functions, maximum response-times, memory limits and output format. The paper will also include considerations about what quantitative improvements AMIDST should obtain over state of the art.

In section 2, AMIDST relevant methodologies for evaluation of both batch and streaming algorithms are identified and discussed. This section forms the foundation of the subsequent sections, where the exact evaluation routines for each use case provider is given. These sections contains a description of the requirements related to evaluation as described in Delivery 1.2, a short description of the algorithms and the data and finally methods for evaluating predictive and runtime performances. Section 6 concludes the report.

2 Test and evaluation methodology

Here we should cover general methods for doing test and evaluation of models in a streaming context. These methods will subsequently be instantiated in relation to the three use case providers so I guess that we should primarily consider the methods that are directly related to the needs of the use case providers, but (taking the back ground of one of the reviewers into account) we might probably benefit from going a bit beyond the immediate needs and put all this stuff into a broader context ...

There has already been some work in this context. A quick search with Google produced the following. I haven't looked at the papers in any great detail, but considering the titles and authors they certainly seems relevant.

[1], [2], [3], [4],

Kaptein, Maurits. 2014. "RStorm: Developing and Testing Streaming Algorithms in R." *Journal.r-Project.org* 6: 123-132. Accessed November 18. <http://journal.r-project.org/archive/accepted/kaptein.pdf>.

João Gama and Raquel Sebastião and Pedro Pereira Rodrigues, Issues in Evaluation of

Stream Learning Algorithms.

Gama, J, PP Rodrigues, and R Sebastião. 2009. “Evaluating Algorithms That Learn from Data Streams.” <http://dl.acm.org/citation.cfm?id=1529616>.

Gama, João, Raquel Sebastião, and Pedro Pereira Rodrigues. 2012. “On Evaluating Stream Learning Algorithms.” *Machine Learning* 90 (3) (October 24): 317-346. doi:10.1007/s10994-012-5320-9. <http://link.springer.com/10.1007/s10994-012-5320-9>.

3 Cajamar: test and evaluation

3.1 Use case requirements

Summarize the use case requirements for the different application scenarios. This information should be derived from Deliverable 1.2.

3.2 Model and data characteristics

Describe aspects of the model and data relevant for the ensuing test and evaluation discussion. Much of this information can be synthesized from the existing documents, and should serve to make the document more self-contained.

3.3 Predictive performance: test and evaluation

3.3.1 Application scenario 1

3.3.2 Application scenario 2

3.4 Run-time performance: test and evaluation

3.4.1 Application scenario 1

3.4.2 Application scenario 2

4 Daimler: test and evaluation

4.1 Use case requirements

Summarize the use case requirements for the different application scenarios. This information should be derived from Deliverable 1.2.

4.2 Model and data characteristics

Describe aspects of the model and data relevant for the ensuing test and evaluation discussion. Much of this information can be synthesized from the existing documents, and should serve to make the document more self-contained.

4.3 Predictive performance: test and evaluation

4.3.1 Application scenario 1

4.3.2 Application scenario 2

4.4 Run-time performance: test and evaluation

4.4.1 Application scenario 1

4.4.2 Application scenario 2

5 Verdande: test and evaluation

5.1 Use case requirements

Summarize the use case requirements for the different application scenarios. This information should be derived from Deliverable 1.2.

5.2 Model and data characteristics

Describe aspects of the model and data relevant for the ensuing test and evaluation discussion. Much of this information can be synthesized from the existing documents, and should serve to make the document more self-contained.

5.3 Predictive performance: test and evaluation

5.3.1 Application scenario 1

5.3.2 Application scenario 2

5.4 Run-time performance: test and evaluation

5.4.1 Application scenario 1

5.4.2 Application scenario 2

6 Conclusion

References

- [1] Kaptein, M.: Rstorm: Developing and testing streaming algorithms in r. *The R Journal* **6**(1) (2014) 123–132
 - [2] Gama, J.a., Rodrigues, P.P., Sebastião, R.: Evaluating algorithms that learn from data streams. In: *Proceedings of the 2009 ACM Symposium on Applied Computing. SAC '09*, New York, NY, USA, ACM (2009) 1496–1500
 - [3] Gama, J.a., Sebastião, R., Rodrigues, P.P.: Issues in evaluation of stream learning algorithms. In: *15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*. (2009) 329–337
 - [4] Gama, J., Sebastião, R., Rodrigues, P.P.: On evaluating stream learning algorithms. *Machine Learning* **90**(3) (2012) 317–346
-