IJCAI-16 (author)

Help Log out

New Submission | Submission 2164 | IJCAI-16 | Support | News | EasyChair

IJCAI-16 Submission 2164

Update authors
Review response

Submission information updates are disabled.

Click "Review response" to **see the reviews sent to you for the review response period and your response**.

For all questions related to processing your submission you should contact the conference organizers. Click here to see information about this conference.

All **reviews sent to you** can be found at the bottom of this page.

Paper 2164	
Title:	Learning Multi-Step Predictive State Representations
Paper	
Track:	IJCAI-16 Main Track
Author keywords:	Time series prediction Predictive state representations Spectral learning
EasyChair keyphrases:	predictive state representation (142), dynamical system (70), model size (70), spectral learning algorithm (63), double loop environment (63), learning algorithm (60), transition operator (60), multiple observation (50), double loop (50), reduced model size (47), generic coding function (47), basic observation (40), spectral algorithm (40), neural information processing system (40), hankel matrix (40)
Topics:	Contribution Type::Primarily to a single area of AI, Evaluation Methodology::Mixed theoretical/empirical, Machine Learning::Time-series/Data Streams, Machine Learning::Unsupervised Learning
Abstract:	Recent years have seen the development of efficient and provably correct spectral algorithms for learning models of partially observable environments arising in many applications. But despite the high hopes raised by this new class of algorithms, their practical impact is still below expectations. One reason for this is the difficulty in adapting spectral methods to exploit structural constraints about different target environments which can be known beforehand. A natural structure intrinsic to many dynamical systems is a multi-resolution behaviour where interesting phenomena occur at different time scales during the evolution of the system. In this paper we introduce the multi-step predictive state representation (M-PSR) and an associated learning algorithm that finds and leverages frequent patterns of observations at multiple scales in dynamical systems with discrete observations. We perform experiments on robot exploration tasks in a wide variety of environments and conclude that the use of M-PSR improves over the classical PSR for varying amounts of data, environment sizes, and number of observations symbols.
Time:	Jan 27, 16:56 GMT
Technical Track	(Main) The LEAD author of this submission is a (graduate or undergraduate) student (Main) At least one of the authors of this submission is an undergraduate student
(Required only for the technical track) Confirm that your submission follows IJCAI-16 requirements	I confirm that this paper is not currently under review at any other archival venue, and will not be submitted to any such venue until the end of IJCAI reviewing period (4/4/2016) I confirm that I have followed all the instructions about anonymizing the paper to support IJCAI double-blind reviewing. I confirm that the paper is not longer than 7 pages, with the 7th page containing nothing other than references I confirm that the paper has been formatted in IJCAI two-column, camera-ready style with US letter size; and Type 1 or True Type fonts. I understand that papers violating IJCAI submission rules are subject to automatic rejection without review.

Authors

first name	last name	email	country	organization	Web page	corresponding?
Lucas	Langer	lucaslangermont9@gmail.com	Canada	McGill University		
Borja	Balle	b.deballepigem@lancaster.ac.uk	United Kingdom	Lancaster University		
Doina	Precup	dprecup@cs.mcgill.ca	Canada	McGill University, School of Computer Science	http://www.cs.mcgill.ca/~dprecup	✔

Reviews

Review 2

Significance:	**2**: (medium (modest contribution or average impact))
Soundness:	**3**: (correct)
Scholarship:	**3**: (excellent coverage of related work)
Clarity:	**3**: (well written)
Breadth of Interest:	**2**: (limited to specialty area)
Summary Rating:	**3**: (+++)
Confidence:	**2**: (reasonably confident)
Summarize the Main Contribution of the Paper:	This work presents the multi-step predictive state representation (M-PSR) which augments the canonical PSR with a set of multi-step observations, a coding function, and a modified set of transition matrices. The authors describe a spectral learning algorithm for learning data-driven M-PSRs using Hankel matrices. Experimental results on two simulation environments (Double Loop and Pacman) show that the M-PSR outperforms the PSR in terms of norm difference between the estimated and true probability distributions.

Comments for the Authors:

== Revised Review ==
Thank you to the authors for responding to my queries regarding the basis choices and compression, and for revising the paper with running time information. My initial assessment still stands; I think the paper is interesting, advances PSRs in a significant manner, and should be accepted.

== Initial Review ==
Overall, the paper is well-written and describes an interesting extension to PSRs. Sections 1 and 2 provide a good background and motivate the problem well. Although the experiments are limited to two "toy" problems (plus one modification), they serve to illustrate the utility of the representation (particularly for small model sizes), and that the learning algorithm manages to identify the "correct" operators from the training data.

A few comments I think will help readability:

- Sec 3.1: A brief statement on choosing P and S would be helpful (and make the paper more self-contained). From my reading, the choice of these sets can have a significant impact on the representation? A natural question is how sensitive the representation is to "poor" basis choices?

- Sec 3.4: Is the greedy scheme optimal in any sense? More generally, can any suitable compression method be substituted in?

- Sec 4: I couldn't find information related to how long it took to learn the M-PSRs, relative to the typical PSR. A concern is the scalability of this approach to large-scale problems involving very large state and observation spaces. Can the authors comment?

- Sec 4: Figure 5:, the panels on right (stretch factor) have different model size range from the left. Just to clarify, were the ranges truncated because the M-PSR and the PSR performed similarly beyond 40?

- Sec 5: mentions briefly potential computational gains when performing conditional probability queries since M-PSR uses smaller matrices. This is certainly a nice feature of having a compressed representation; can the authors give some quantitative timing results of the difference in computational times in the domains studied?

Minor comments:

- Sec 3.3, 1st par, line 2, "observations"->"observation"

- Sec 5, left col, last line, this sentence (which continues on to the right column) is grammatically incorrect.

- Fig 6, caption, "(left"->"(left)"

- I found the graphs a little hard to read. Given the page limit, I can empathize, but I think Fig 2 right (the Pacman graph) could be replotted to reduce height, which would give some additional space to increase the size of the very small figures.

Review 1

Significance:	**3**: (high (substantial contribution or strong impact))
Soundness:	**3**: (correct)
Scholarship:	**2**: (relevant literature cited but could expand)
Clarity:	**3**: (well written)
Breadth of Interest:	**3**: (some interest beyond specialty area)
Summary Rating:	**4**: (++++)
Summarize the Main Contribution of the Paper:	The paper proposed Multi-step Predictive State Presentation (M-PSR), an extension to the standard PSR model aiming at handle frequent patterns occuring at multiple time scales, which subsumes the previous timing model as its special case. This is analogues in spirit to the option framework in reinforcement learning, and similar gains are expected in the context of learning models of dynamical systems. A detailed model construction procedure is developed. Empirical evaluations demonstrate the effectiveness of the model compared to standard approaches, and the learned operators are found to capture the nature of the domains. The paper is well motivated, clearly presented, and supported by detailed empirical results. Improving the practicality of PSR (esp. with spectral learning) has been of substantial interest in the community, and one of the reasons for PSR not working well in practice, as argued by the paper, is the lack of ability to incorporate domain structure, and this paper considers a particular structure of multiple scale of events.
Comments for the Authors:	In general I like this paper. Still, I hope the author(s) could elaborate more on the following question to help me (and other readers) understand the strength and value of M-PSR: Why do we want to use M-PSR? Yes, you have experiments where M-PSR outperforms standard PSR. But can you provide a simple, sharp, and potentially theoretical explanation for that? We could include strings of various lengths in the basis of standard PSR, and why is that not enough to capture the multiple timing scale structure of a problem? (My guess is that PSR predictions suffer from the issue of errors cumulating over multiple matrix multiplications, but I am not sure.) I notice that you cited the option framework in RL as an analogy; regarding options, we do have a simple yet fundamental explanation on why it's useful: using options shrinks the discount factor. Can you provide something similar here, and maybe accompany by a minimal running example? Minor issues - The example on Base M-PSR in Sec 2.1: shouldn't b be such that b > 1? - Figures are too small (need something like 300% zoom to see on the screen, and no way to see when printed out). - Do standard PSR and M-PSR share the same basis in the experiment? - It would be interesting to see experiment results on natural language domains. For example, in character-level language modelling, would M-PSR learn something like common word prefix/suffix /root?

Review 1

Significance:	**3**: (high (substantial contribution or strong impact))
Soundness:	**3**: (correct)
Scholarship:	**2**: (relevant literature cited but could expand)
Clarity:	**3**: (well written)
Breadth of Interest:	**3**: (some interest beyond specialty area)
Summary Rating:	**4**: (++++)

Confidence:	**2**: (reasonably confident)

Summarize the Main Contribution of the Paper:	The paper proposed Multi-step Predictive State Presentation (M-PSR), an extension to the standard PSR model aiming at handle frequent patterns occuring at multiple time scales, which subsumes the previous timing model as its special case. This is analogues in spirit to the option framework in reinforcement learning, and similar gains are expected in the context of learning models of dynamical systems. A detailed model construction procedure is developed. Empirical evaluations demonstrate the effectiveness of the model compared to standard approaches, and the learned operators are

Comments for the Authors:	found to capture the nature of the domains. Thanks for the response. It's great to hear that you will add examples to illustrate how M-PSR helps. And, one more comment: for the advantages of M-PSR in the reduced rank setting, I thought low-rankness will cause approximation error in learning hence contribute to the error in the update matrices (together with the estimation error from the finite sample effect), and then such errors cumulate in multiplications so reducing the number of multiplications by M-PSR helps. The author(s) seem indicating a different reason besides this one: that is, given a fixed model rank (which is potentially very low), you could possibly approximate a problem much better with M-PSR than with standard PSR? If that is true it will be very interesting, and the author(s) should explain such an advantage clearly in the paper. -----------initial review below------------- The paper is well motivated, clearly presented, and supported by detailed empirical results. Improving the practicality of PSR (esp. with spectral learning) has been of substantial interest in the community, and one of the reasons for PSR not working well in practice, as argued by the paper, is the lack of ability to incorporate domain structure, and this paper considers a particular structure of multiple scale of events. In general I like this paper. Still, I hope the author(s) could elaborate more on the following question to help me (and other readers) understand the strength and value of M-PSR: Why do we want to use M-PSR? Yes, you have experiments where M-PSR outperforms standard PSR. But can you provide a simple, sharp, and potentially theoretical explanation for that? We could include strings of various lengths in the basis of standard PSR, and why is that not enough to capture the multiple timing scale structure of a problem? (My guess is that PSR predictions suffer from the issue of errors cumulating over multiple matrix multiplications, but I am not sure.) I notice that you cited the option framework in RL as an analogy; regarding options, we do have a simple yet fundamental explanation on why it's useful: using options shrinks the discount factor. Can you provide something similar here, and maybe accompany by a minimal running example? Minor issues - The example on Base M-PSR in Sec 2.1: shouldn't b be such that b > 1? - Figures are too small (need something like 300% zoom to see on the screen, and no way to see when printed out). - Do standard PSR and M-PSR share the same basis in the experiment? - It would be interesting to see experiment results on natural language domains. For example, in character-level language modelling, would M-PSR learn something like common word prefix/suffix /root?

Review 3

Significance:	**2**: (medium (modest contribution or average impact))
Soundness:	**3**: (correct)
Scholarship:	**2**: (relevant literature cited but could expand)
Clarity:	**2**: (mostly readable with some scope of improvement)
Breadth of Interest:	**3**: (some interest beyond specialty area)
Summary Rating:	**2**: (++)
Confidence:	**3**: (highly confident)
Summarize the Main Contribution of the Paper:	This paper presents an interesting modification to spectral PSRs, by introducing the idea of multi-step update functions (matrices). This is particularly useful when developing system models that have state representation that is more compact than the true model (e.g. has many fewer entries in the state vector than are needed for a complete representation). The stated motivation for this is that the multi-step updates can implicitly contain updates in a compact form that capture the contributions of each step. This motivation, in fact, is the key advantage to using these multi-step PSRs. The paper develops the multi-step update algorithms and analyzes the algorithm on a few mid-sized problems. However there is no theoretical contribution, the methods are only verified empirically. The paper is generally clear and well-written, although it would benefit from clearly stating up front that the goal was to develop multi-step state updates, rather than multi-step predictions (which already exist in traditional PSR formulations). The empirical investigation certainly suggests that benefits are possible by using the M-PSR over PSRs, especially on compact representations. However, further investigation on a larger range of problems would be necessary before that statement could be broadly applied. One other area that is somewhat troubling is that a state update must be learned for each of the

	multi-step sequences. Therefore, this representation does increase model size, and thus also increase the amount of required training data (especially because the training data for a given multi-step sequence must correspond to that particular multi-step sequence occurring in the data). There are two areas that would benefit from further theoretical investigation.
Comments for the Authors:	First, an analysis of the connection between many single-step updates and a single multi-step update. Some sort of approximation accuracy analysis for these compact updates would strengthen the paper considerably. For instance, a compact representation could be projected to a full representation, then many single-step updates applied, followed by projection back to the compact repn. This could be compared to the compact multi-step update. Second, if I understand correctly, the state representation at time n may correspond to a completely different sequence of updates than that at time n+1. In other words, it is not simply a matter of updating state n to get state n+1. Understanding the consistency of state representation over time is also interesting. Because the single-step observations are in \Sigma', you could apply the traditional update to state n, and compare that to the M-PSR state update. This would give an idea of variability within this representation.

Review 2

Significance:	2: (medium (modest contribution or average impact))
Soundness:	3: (correct)
Scholarship:	3: (excellent coverage of related work)
Clarity:	3: (well written)
Breadth of Interest:	2: (limited to specialty area)
Summary Rating:	3: (+++)
Summarize the Main Contribution of the Paper:	This work presents the multi-step predictive state representation (M-PSR) which augments the canonical PSR with a set of multi-step observations, a coding function, and a modified set of transition matrices. The authors describe a spectral learning algorithm for learning data-driven M-PSRs using Hankel matrices. Experimental results on two simulation environments (Double Loop and Pacman) show that the M-PSR outperforms the PSR in terms of norm difference between the estimated and true probability distributions.
Comments for the Authors:	Overall, the paper is well-written and describes an interesting extension to PSRs. Sections 1 and 2 provide a good background and motivate the problem well. Although the experiments are limited to two "toy" problems (plus one modification), they serve to illustrate the utility of the representation (particularly for small model sizes), and that the learning algorithm manages to identify the "correct" operators from the training data. A few comments I think will help readability: - Sec 3.1: A brief statement on choosing P and S would be helpful (and make the paper more self-contained). From my reading, the choice of these sets can have a significant impact on the representation? A natural question is how sensitive the representation is to "poor" basis choices? - Sec 3.4: Is the greedy scheme optimal in any sense? More generally, can any suitable compression method be substituted in? - Sec 4: I couldn't find information related to how long it took to learn the M-PSRs, relative to the typical PSR. A concern is the scalability of this approach to large-scale problems involving very large state and observation spaces. Can the authors comment? - Sec 4: Figure 5:, the panels on right (stretch factor) have different model size range from the left. Just to clarify, were the ranges truncated because the M-PSR and the PSR performed similarly beyond 40? - Sec 5: mentions briefly potential computational gains when performing conditional probability queries since M-PSR uses smaller matrices. This is certainly a nice feature of having a compressed representation; can the authors give some quantitative timing results of the difference in computational times in the domains studied? Minor comments: - Sec 3.3, 1st par, line 2, "observations"->"observation" - Sec 5, left col, last line, this sentence (which continues on to the right column) is grammatically incorrect.

– Fig 6, caption, "(left"->"(left)"

– I found the graphs a little hard to read. Given the page limit, I can empathize, but I think Fig 2 right (the Pacman graph) could be replotted to reduce height, which would give some additional space to increase the size of the very small figures.