Regularized Contextual Policy Search via Mutual Information

Simone Parisi
Technical University of Darmstadt
64289 Darmstadt, Germany
parisi@ias.tu-darmstadt.de

Voot Tangkaratt
The University of Tokyo
113-0033 Tokyo, Japan
voot@ms.k.u-tokyo.ac.jp

Jan Peters
Technical University of Darmstadt
64289 Darmstadt, Germany
mail@jan-peters.net

Abstract

Contextual policy search algorithms are black-box optimizers that learn to improve policy parameters and simultaneously generalize these parameters to different context or task variables. However, defining a context representation on which policy search can perform well is a tedious but crucial process. It typically requires expert knowledge, does not generalize straightforwardly over different tasks and strongly influences the quality of the learned policy. Furthermore, existing algorithms usually perform dimensionality reduction taking into account only feature redundancy and relevance, ignoring the problem of feature interaction. In this paper, we present an autonomous feature construction algorithm for learning low-dimensional manifolds of goal-relevant features jointly with an optimal policy. We learn a model of the reward that is locally quadratic in both the policy parameters and the context variables. To tackle high dimensional context variables and to take into account feature interaction, we propose to regularize the model by mutual information.

Keywords: reinforcement learning, policy search, mutual information, model-based, relative entropy, kullback-leibler divergence

Acknowledgements

This work was funded by a DFG grant within the priority program “Autonomous learning” (SPP1527).

1 Introduction

An autonomous agent often requires different policies for solving tasks with different contexts. For instance, in a ball hitting task the robot has to adapt his controller according to the ball position, i.e., the context. Contextual policy search approaches [11, 5, 4] represent the contexts by real-valued vectors and are able to learn a context-dependent distribution over the policy parameters. Such a distribution can generalize across context values and therefore the agent is able to adapt to unseen contexts. However, these algorithms rely on concise and dense, but informative, context descriptions. Complex tasks often result in large sparse spaces. Such high-dimensional representations can contain features which are redundant or even irrelevant for both the description of the context and the goal of the agent. Furthermore, two or more weak features could explain the output well in the context of each other, even though each of them alone may not be explanatory. Scaling up contextual policy search by learning a parsimonious context representation is thus an important step towards more autonomous and widely applicable algorithms.

However, learning features for policy search algorithms jointly with an optimal policy introduces a circular dependency into the learning process. To learn goal-relevant features a large part of the context space needs to be explored using a goal-achieving policy. At the same time, learning a locally or globally optimal policy requires a concise feature representation. For this reason, common RL approaches separate the problem of learning the context representation from the agent task by preprocessing the feature space [2, 12]. As it precedes the RL procedure, such a separated dimensionality reduction step requires additional context samples that are frequently not helpful for the learning itself. The context representation resulting from a preceding dimensionality reduction step, in fact, does not consider the importance of features for optimal policies but only for general policies. Thus, the context representation will not optimally support the RL algorithm as it may establish features that are only relevant for suboptimal or even poor policies.

Recently, Tangkaratt et al. [18] proposed an algorithm to overcome these limitations, namely Contextual Model-based Relative Entropy Stochastic Search (C-MORE). C-MORE learns a surrogate quadratic model of the reward to compute a Gaussian search distribution analytically. To balance the exploration-exploitation trade-off, the algorithm upper-bounds the Kullback-Leibler divergence between successive search distributions and lower-bounds the entropy of the new search distribution. Yet, the authors focused only on feature redundancy and investigated only nuclear norm regularization.

In the field of machine learning, mutual information (MI), and especially squared-loss mutual information (SLMI), have increasingly become important and popular. In information theory, MI and SLMI are quantities measuring statistical dependency between random variables [3]. They both can be used to detect non-linear dependencies between two random variables and have been widely used for dimensionality reduction [19, 14, 17, 13]. However, unlike maximizing the MI, SLMI regularization is strictly convex under mild conditions [9] and therefore more appealing. SLMI has also been successfully applied in problems of feature interaction, i.e., when a variable strongly depends on the interaction of two or more features, but not directly on each feature alone.

In this paper, we present the C-MORE framework for contextual RL problems and discuss the its extension with SLMI.

2 Problem Statement and Notation

2.1 Problem Formulation

The contextual policy search is formulated as follows. An agent observes the context variable $\boldsymbol{c} \in \mathbb{R}^{d_c}$ and draws a parameter $\theta \in \mathbb{R}^{d_\theta}$ from a search distribution $p(\theta|\boldsymbol{c})$. Subsequently, the agent executes a policy with the parameter θ and observes a scalar reward computed by a reward function $R(\theta, \boldsymbol{c})$. The goal is to find a search distribution $p(\theta|\boldsymbol{c})$ maximizing the expected reward

$$\iint \mu(\boldsymbol{c}) p(\theta|\boldsymbol{c}) R(\theta, \boldsymbol{c}) \, \mathrm{d}\theta \, \mathrm{d}\boldsymbol{c}, \tag{1}$$

where $\mu(\boldsymbol{c})$ denotes the context distribution. We assume that the reward function $R(\theta, \boldsymbol{c})$ itself is unknown, but the agent can always access the reward value. We stress that context variables are fixed during task execution and they are drawn independently from $\mu(\boldsymbol{c})$. Thus, context variables are different from state variables in standard policy search.

2.2 Related Work

In the basic contextual policy search framework, the agent iteratively collects samples $\{(\theta^{[i]}, \boldsymbol{c}^{[i]}, R(\theta^{[i]}, \boldsymbol{c}^{[i]}))\}_{i=1}^{N}$ using a sampling distribution $q(\theta|\boldsymbol{c})$. Subsequently, it computes a new search distribution $p(\theta|\boldsymbol{c})$ such that the expected reward increases or is maximized. In literature, different approaches have been used to compute the new search distribution, e.g., evolutionary strategies [8], expectation-maximization algorithms [11], or information theoretic approaches [5].

Most of the existing contextual policy search methods focus on tasks with low-dimensional context variables. If the context variable space is high-dimensional, usually the problem of learning a low-dimensional context representation is separated from the policy search by preprocessing the context space. However, unsupervised linear dimensionality reduction techniques are insufficient in problems where the latent representation contains distractor dimensions that do not influence the reward, or where the reward is explained by the interaction of two or more features. A prominent example is principal component analysis (PCA) [10], that does not take the supervisory signal into account and therefore cannot discriminate between relevant and irrelevant latent dimensions and cannot detect feature interactions. On the other hand, supervised linear dimensionality reduction techniques require a suitable response variable. However, manually defining such variables in nontrivial for many problems.

In recent years, non-linear dimensionality reduction techniques based on deep learning have gained popularity [1] For instance, Watter et al. [21] proposed a generative deep network to learn low-dimensional representations of images in order to capture information about the system transition dynamics and allow optimal control problems to be solved in low-dimensional spaces. More recently, Silver et al. [16] successfully trained a machine to play a high-level game of *go* using a deep convolutional network. Although their work does not directly focus on dimensionality reduction, deep convolutional networks are known to be able to extract meaningful data representations. Thus, the effect of dimensionality reduction is achieved. However, deep learning approaches generally require large datasets that are difficult to obtain in real-world scenarios (e.g., robotics). Furthermore, they involve solving non-convex optimization, which can suffer from local optima.

To alleviate these issues, Tangkaratt et al. [18] proposed to learn a low-rank representation of the reward function by nuclear norm regularization. They show that learning a low-rank representation corresponds to performing linear dimensionality reduction on the context variables. Since optimization with a rank constraint is NP-hard, they minimize the nuclear norm, which is a convex surrogate of the rank function [15].

3 Contextual Model-based Relative Entropy Stochastic Search

C-MORE [18] finds a contextual search distribution maximizing the expected reward while upper-bounding the Kullback-Leibler (KL) divergence between successive search distributions and lower-bounding the entropy of the new search distribution. The KL and the entropy bounds control the exploration-exploitation trade-off. The key insight of C-MORE is to learn a reward model to efficiently compute a new search distribution in closed form.

3.1 Learning the Search Distribution

The goal of C-MORE is to find a search distribution $p(\theta|\boldsymbol{c})$ that maximizes the expected reward while upper-bounding the expected KL divergence between $p(\theta|\boldsymbol{c})$ and $q(\theta|\boldsymbol{c})$, and lower-bounding the expected entropy of $p(\theta|\boldsymbol{c})$. Formally,

$$\begin{aligned}
\max_p & \iint \mu(\boldsymbol{c})p(\theta|\boldsymbol{c})R(\theta,\boldsymbol{c})\,\mathrm{d}\theta\,\mathrm{d}\boldsymbol{c},\\
\text{s.t.} & \iint \mu(\boldsymbol{c})p(\theta|\boldsymbol{c})\log\frac{p(\theta|\boldsymbol{c})}{q(\theta|\boldsymbol{c})}\,\mathrm{d}\theta\,\mathrm{d}\boldsymbol{c} \leq \epsilon,\\
- & \iint \mu(\boldsymbol{c})p(\theta|\boldsymbol{c})\log p(\theta|\boldsymbol{c})\,\mathrm{d}\theta\,\mathrm{d}\boldsymbol{c} \geq \beta,\\
& \iint \mu(\boldsymbol{c})p(\theta|\boldsymbol{c})\,\mathrm{d}\theta\,\mathrm{d}\boldsymbol{c} = 1,
\end{aligned}$$

where the KL upper-bound ϵ and the entropy lower-bound β are parameters specified by the user. The former is fixed for the whole learning process. The latter is adaptively changed according to the percentage of the relative difference between the sampling policy's expected entropy and the minimal entropy, i.e.,

$$\beta = \gamma(\mathbb{E}[H(q)] - H_0) + H_0,$$

where $\mathbb{E}[H(q)] = -\iint \mu(\boldsymbol{c})q(\theta|\boldsymbol{c})\log q(\theta|\boldsymbol{c})\,\mathrm{d}\theta\,\mathrm{d}\boldsymbol{c}$ is the sampling policy's expected entropy and H_0 is the minimal entropy. The above optimization problem can be solved by the method of Lagrange multipliers. The solution is given by

$$p(\theta|\boldsymbol{c}) = q(\theta|\boldsymbol{c})^{\frac{\eta}{\eta+\omega}}\exp\left(\frac{R(\theta,\boldsymbol{c})}{\eta+\omega}\right)\exp\left(-\frac{\eta+\omega-\gamma}{\eta+\omega}\right). \tag{2}$$

The Lagrange multipliers $\eta > 0$ and $\omega > 0$ are obtained by minimizing

$$g(\eta,\omega) = \eta\epsilon - \omega\beta + (\eta+\omega)\int \mu(\boldsymbol{c})\left(\log\int q(\theta|\boldsymbol{c})^{\frac{\eta}{\eta+\omega}}\exp\left(\frac{R(\theta,\boldsymbol{c})}{\eta+\omega}\right)\mathrm{d}\theta\right)\mathrm{d}\boldsymbol{c}. \tag{3}$$

Evaluating the dual is not trivial due to the integration over $q(\theta|\boldsymbol{c})^{\frac{\eta}{\eta+\omega}}$, that cannot be approximated straightforwardly by sample averages. Below, we describe how to solve this issue and evaluate the dual function from data.

3.2 Analytical Solution via the Quadratic Model

We assume that the reward function $R(\theta, \boldsymbol{c})$ can be approximated by a quadratic model

$$\widehat{R}(\theta, \boldsymbol{c}) = \theta^\mathsf{T} \boldsymbol{A} \theta + \boldsymbol{c}^\mathsf{T} \boldsymbol{B} \boldsymbol{c} + 2\theta^\mathsf{T} \boldsymbol{D} \boldsymbol{c} + \theta^\mathsf{T} \boldsymbol{r}_1 + \boldsymbol{c}^\mathsf{T} \boldsymbol{r}_2 + r_0, \tag{4}$$

where $\boldsymbol{A} \in \mathbb{R}^{d_\theta \times d_\theta}, \boldsymbol{B} \in \mathbb{R}^{d_c \times d_c}, \boldsymbol{D} \in \mathbb{R}^{d_\theta \times d_c}, \boldsymbol{r}_1 \in \mathbb{R}^{d_\theta}, \boldsymbol{r}_2 \in \mathbb{R}^{d_c}$, and $r_0 \in \mathbb{R}$ are the model parameters. Matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ are symmetric. We also assume the sampling distribution $q(\theta|\boldsymbol{c})$ to be Gaussian of the form

$$q(\theta|\boldsymbol{c}) = \mathcal{N}(\theta|\boldsymbol{B} + \boldsymbol{K}\boldsymbol{c}, \boldsymbol{Q}). \tag{5}$$

Under these assumptions, the dual function in Eq. (3) can be expressed as

$$g(\eta,\omega) = \eta\epsilon - \omega\beta + \frac{1}{2}\Big(\boldsymbol{f}^\mathsf{T} \boldsymbol{F}^{-1} \boldsymbol{f} - \eta \boldsymbol{B}^\mathsf{T} \boldsymbol{Q}^{-1} \boldsymbol{B} + (\eta+\omega)\log|2\pi \boldsymbol{F}^{-1}(\eta+\omega)| - \eta \log|2\pi \boldsymbol{Q}|\Big) + \int \mu(\boldsymbol{c}) \Big(\boldsymbol{c}^\mathsf{T} \boldsymbol{m} + \frac{1}{2}\boldsymbol{c}^\mathsf{T} \boldsymbol{M} \boldsymbol{c}\Big) \mathrm{d}\boldsymbol{c}, \tag{6}$$

where

$$\begin{aligned}
\boldsymbol{f} &= \eta \boldsymbol{Q}^{-1} \boldsymbol{B} + \boldsymbol{r}_1, \\
\boldsymbol{F} &= \eta \boldsymbol{Q}^{-1} - 2\boldsymbol{A}, \\
\boldsymbol{m} &= \boldsymbol{L}^\mathsf{T} \boldsymbol{F}^{-1} \boldsymbol{f} - \eta \boldsymbol{K}^\mathsf{T} \boldsymbol{Q}^{-1} \boldsymbol{B}, \\
\boldsymbol{M} &= \boldsymbol{L}^\mathsf{T} \boldsymbol{F}^{-1} \boldsymbol{L} - \eta \boldsymbol{K}^\mathsf{T} \boldsymbol{Q}^{-1} \boldsymbol{K}, \\
\boldsymbol{L} &= \eta \boldsymbol{Q}^{-1} \boldsymbol{K} + 2\boldsymbol{D}.
\end{aligned}$$

Since the context distribution $\mu(\boldsymbol{c})$ is unknown, we approximate the expectation in Eq. (6) by sample averages. The dual function can be minimized by standard non-linear optimization routines. Finally, using Eq. (4) and Eq. (5) the new search distribution $p(\theta|\boldsymbol{c})$ is computed in closed form as

$$p(\theta|\boldsymbol{c}) = \mathcal{N}\Big(\theta|\boldsymbol{F}^{-1}\boldsymbol{f} + \boldsymbol{F}^{-1}\boldsymbol{L}\boldsymbol{c}, \boldsymbol{F}^{-1}(\eta+\omega)\Big). \tag{7}$$

To ensure that the covariance $\boldsymbol{F}^{-1}(\eta+\omega)$ is positive definite, the matrix $\boldsymbol{A}$ of the quadratic model is constrained to be negative definite.

3.3 Linear Dimensionality Reduction on the Context Variables

Linear dimensionality reduction learns a low-rank matrix $\boldsymbol{W}$ and projects the data onto a lower dimensional subspace. Performing linear dimensionality reduction on the context variables yields the following quadratic model

$$\widehat{R}(\theta, \boldsymbol{c}) = \theta^\mathsf{T} \boldsymbol{A} \theta + \boldsymbol{c}^\mathsf{T} \boldsymbol{W}^\mathsf{T} \widetilde{\boldsymbol{B}} \boldsymbol{W} \boldsymbol{c} + 2\theta^\mathsf{T} \widetilde{\boldsymbol{D}} \boldsymbol{W} \boldsymbol{c} + \theta^\mathsf{T} \boldsymbol{r}_1 + \boldsymbol{c}^\mathsf{T} \boldsymbol{W}^\mathsf{T} \widetilde{\boldsymbol{r}}_2 + r_0, \tag{8}$$

where $\boldsymbol{W} \in \mathbb{R}^{d_z \times d_c}$ denotes a rank-d_z matrix with $d_z < d_c$. The model parameters $\boldsymbol{A}, \widetilde{\boldsymbol{B}}, \widetilde{\boldsymbol{D}}, \boldsymbol{r}_1, \widetilde{\boldsymbol{r}}_2$ and r_0 can be learned by ridge regression. However, the matrix $\boldsymbol{B} = \boldsymbol{W}^\mathsf{T} \widetilde{\boldsymbol{B}} \boldsymbol{W}$ is low-rank, i.e., $\operatorname{rank}(\boldsymbol{B}) = d_z < d_c$. Thus, performing linear dimensionality reduction on the contexts makes $\boldsymbol{B}$ low-rank. Note that the rank of $\boldsymbol{D} = \widetilde{\boldsymbol{D}} \boldsymbol{W}$ depends on θ and is problem dependent. Hence, we do not consider the rank of $\boldsymbol{D}$ for dimensionality reduction.

There are several linear dimensionality reduction methods that can be applied to learn $\boldsymbol{W}$. Principal component analysis (PCA) [10] is a common method used in statistics and machine learning. However, being unsupervised, it does not take the regression targets into account, i.e., the reward. Alternative supervised techniques, such as kernel dimension reduction [7], do not take the regression model, i.e., the quadratic model, into account. On the contrary, in projection regression [6, 20] the model parameters and the projection matrix are learned simultaneously. However, applying this approach to the model in Eq. (8) requires alternately optimizing for the model parameters and the projection matrix and is computationally expensive.

4 Discussion

The performance of C-MORE strongly depends on the accuracy of the quadratic model. The quadratic model can be learned by regression methods such as ridge regression. However, ridge regression is prone to error when the context is high-dimensional. The original C-MORE tackles this issue by nuclear norm regularization on the contextual matrix of the surrogate model. Yet, nuclear norm regularization does not directly take into account possible interactions between features. Therefore, we plan to extend the C-MORE framework with SLMI regularization.

References

[1] Yoshua Bengio. Learning deep architectures for AI. *Foundations and trends in Machine Learning*, 2(1):1–127, 2009.

[2] Sebastian Bitzer, Matthew Howard, and Sethu Vijayakumar. Using dimensionality reduction to exploit constraints in reinforcement learning. In *Proceedings of the International Conference on Intelligent Robots and Systems (IROS)*, pages 3219–3225, 2010.

[3] Thomas M Cover and Joy A Thomas. *Elements of information theory*. John Wiley & Sons, 2012.

[4] Bruno Da Silva, George Konidaris, and Andrew Barto. Learning parameterized skills. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2012.

[5] Marc Peter Deisenroth, Gerhard Neumann, and Jan Peters. A survey on policy search for robotics. *Foundations and Trends in Robotics*, 2(1-2):1–142, 2013.

[6] Jerome H Friedman and Werner Stuetzle. Projection pursuit regression. *Journal of the American Statistical Association*, 76(376):817–823, 1981.

[7] Kenji Fukumizu, Francis R Bach, and Michael I Jordan. Kernel dimension reduction in regression. *The Annals of Statistics*, pages 1871–1905, 2009.

[8] Nikolaus Hansen, Sibylle D Müller, and Petros Koumoutsakos. Reducing the time complexity of the derandomized evolution strategy with covariance matrix adaptation (cma-es). *Evolutionary Computation*, 11(1):1–18, 2003.

[9] Wittawat Jitkrittum, Hirotaka Hachiya, and Masashi Sugiyama. Feature selection via ℓ_1-penalized squared-loss mutual information. *Transactions on Information and Systems*, 96(7):1513–1524, 2013.

[10] Ian Jolliffe. *Principal component analysis*. Wiley Online Library, 2002.

[11] Jens Kober, Andreas Wilhelm, Erhan Oztop, and Jan Peters. Reinforcement learning to adjust parametrized motor primitives to new situations. *Autonomous Robots*, 33(4):361–379, 2012.

[12] Sascha Lange and Martin Riedmiller. Deep auto-encoder neural networks in reinforcement learning. In *Proceedings of the International Joint Conference on Neural Networks (IJCNN)*, pages 1–8, 2010.

[13] Gang Niu, Wittawat Jitkrittum, Bo Dai, Hirotaka Hachiya, and Masashi Sugiyama. Squared-loss mutual information regularization: A novel information-theoretic approach to semi-supervised learning. In *Proceedings of the International Conference on Machine learning (ICML)*, pages 10–18, 2013.

[14] Hanchuan Peng, Fuhui Long, and Chris Ding. Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy. *Transactions on Pattern Analysis and Machine Intelligence*, 27(8): 1226–1238, 2005.

[15] Benjamin Recht, Maryam Fazel, and Pablo A Parrilo. Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization. *SIAM review*, 52(3):471–501, 2010.

[16] David Silver, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, Sander Dieleman1, Dominik Grewe1, John Nham, Nal Kalchbrenner1, Ilya Sutskever, Timothy Lillicrap, Madeleine Leach, Koray Kavukcuoglu, Thore Graepel1, and Demis Hassabis. Mastering the game of go with deep neural networks and tree search. *Nature*, 529(7587): 484–489, 2016.

[17] Taiji Suzuki and Masashi Sugiyama. Sufficient dimension reduction via squared-loss mutual information estimation. *Neural Computation*, 25(3):725–758, 2013.

[18] Voot Tangkaratt, Herke van Hoof, Simone Parisi, Gerhard Neumann, Jan Peters, and Masashi Sugiyama. Policy search with high-dimensional context variables. In *Proceedings of the Conference on Artificial Intelligence (AAAI)*, 2017.

[19] Kari Torkkola. Feature extraction by non-parametric mutual information maximization. *Journal of Machine Learning Research (JMLR)*, 3(Mar):1415–1438, 2003.

[20] Sethu Vijayakumar and Stefan Schaal. Locally weighted projection regression: Incremental real time learning in high dimensional space. In *Proceedings of the International Conference on Machine learning (ICML)*, pages 1079–1086, 2000.

[21] Manuel Watter, Jost Springenberg, Joschka Boedecker, and Martin Riedmiller. Embed to control: A locally linear latent dynamics model for control from raw images. In *Advances in Neural Information Processing Systems (NIPS)*, pages 2746–2754, 2015.