Reinforcement Learning 2014: Tutorial 6 (week from 10. 3. 2014)

These are just a few hints, please do not distribute.

1. Errors in value estimation, errors in policy estimation and "region errors", i.e. has the agent arrived (sufficiently often/at all) in the highly rewarded regions. This corresponds to the three terms in the definition: Q, pi, and mu. Since they are essentially multiplied, there is a chance for a trade-off, at least at sub-optimal solutions. The optimum is usually characterised by degenerate distributions (all probability mass is concentrated on a single state/action). Degeneracy can mean that the parameter dependency is not smooth any more such, i.e. the gradient may not be very useful. It is an advantage of the natural gradient that it transforms (stretches) the parameter space such that the gradient is always defined (however, degenerated cases may result in disvsion of infinity by infinity which is at least practically a problem).

2. Importance sampling was discussed last year, see rl/slides13/rl16.pdf. It can be helpful in practice.

3. There is no general answer here, i.e. it has to be tried out. The more important problem is: When we define the BFs, we do not yet know where the solutions of the RL problem is going to be, i.e. where a high resolution is required. If we have learned something then by changing BFs we may lose what we have learned. The scales and topology of the system may of course be used, in order to define the width and distribution of the BF and their placement. Sometimes the required resolution becomes unexpectedly high: If a pendulum is to be balanced by say two discrete actions, it is difficult to find the exact sequence of fixed-strength pushes from either side such that the pendulum arrives precisely in the center. This sequence will be different for slightly different points, which the BFs cannot easily resolve. When you are away from the center is is less critical as the one of the two pushing action that moves the pendulum in the upwards direction will be required also for nearby states.
   Non-local basis function may sometimes provide a better generalisation to unexplored regions (neural networks often use sigmoids which are non-local BFs).

4. The complexity may be determined by the effective dimension, but we would need to have some pruning or adaptive algorithm to find these, i.e. we are facing a typical dimension reduction task. Again: RL is different from other approaches in machine learning as we don't know the data in advance and the data distribtuion will be different for different stages of the RL process. Evolutionary approaches may help to find better solutions (essentially implying a many restarts of the algorithm). The more relevant question is that the complexity is mainly due to the complexity of the sequence of actions required towards the goal. If these sequences are short or spatially or temporally homogeneous, then the problem is easy even if the state/action space is relatively large.

5. What is relevant here is known as the credit assignment problem. BFs are not necessarily optimal also in this respect. But eligibility traces do not (unlike the look-up table representation) introduce here additional computations, apart from a small factor due to the trace update, but not N-times as much updates.

6. Most realistic problems do require function representations unless you can work on

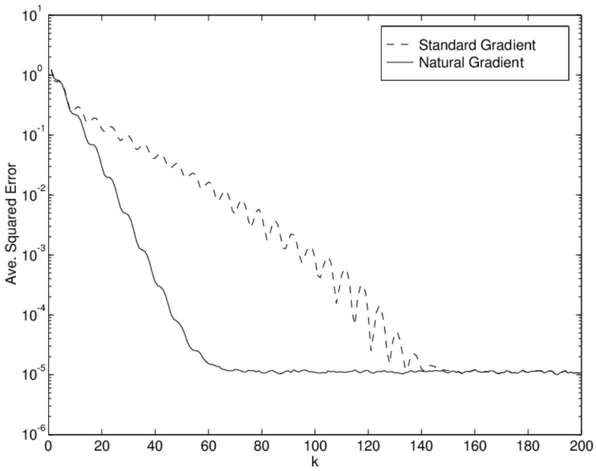
directly on the data (e.g. the REINFORCE algorithms) or if there are e.g. few discrete actions only. What about learning to play chess? What about hybrid representations, where you learn first in the approximation and but later also in the native discretisation of the problem?

7. Solution is to have 50/50 left-right in the two grey fields in inside-pointers in the fields next to the skulls, and obviously a south pointer in the center. So the path can be long but has finite average. The deterministic scheme gets stuck in a portion of all cases. This is a very interesting example, as it shows that stochastic policies can indeed be better than deterministic ones.

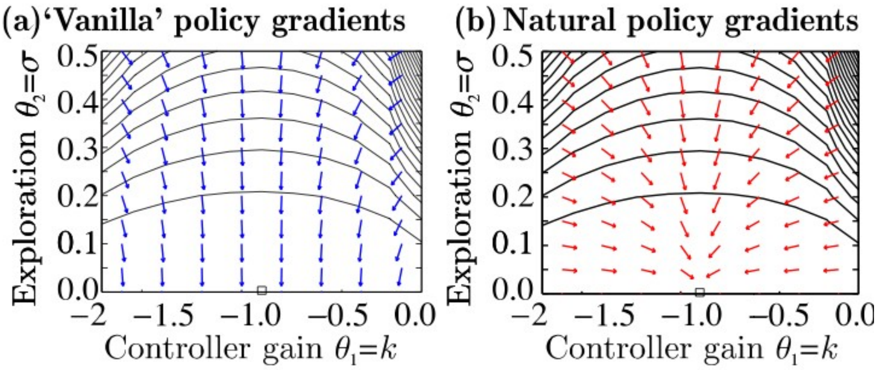
8. phi(x)-psi'(theta), i.e. BF - first moment

9. This was mentioned this in the lecture (just on the whiteboard, therefore is repeated here). The matrix can be decomposed in eigenvectors, which all have positive eigenvalues. Therefore the gradient is still on the correct side of the level line of the function, at least if the learning rate is very small.

10. Here is the result for the stochastic problem from that paper (see arxive.org). Although the problem is counterintuitive w.r.t. the general idea of natural gradients, it is still impressive that the decoupling of the dimensions and the overall more smooth gradient field improves learning in the stochastic case which is mainly by avoiding the oscillations between the regions of different (standard) gradient directions. Note that the improvement can be much larger in more typical from, but even here it seems to work.



Consider also this (less impressive but more appropriate) example how natural gradient works (from Jan Peters' Scholarpedia article on policy gradient.)



Note that the last few problems are not meant as "typical exam questions" but to help a bit e.g. with the understanding of the nAC.