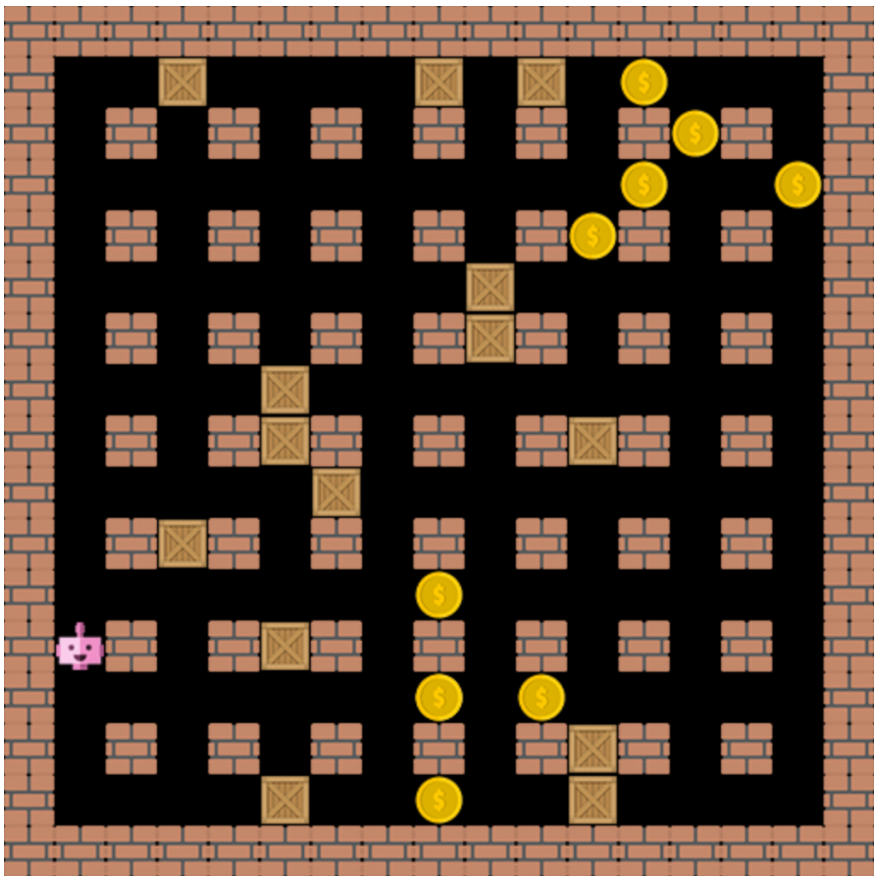
Fundamentals of Machine Learning

Final Project - Report

Reinforcement Learning for Bomberman

Alessandro Motta, Matthias Hericks, Mika Rother

March 28, 2021



Contents

1 Feature Design **3**
1.1 Simple features 3
1.2 Advanced features 3
1.2.1 Shortest path algorithms 3

Chapter 1

Feature Design

1.1 Simple features

by Mika Rother

To see how many features there are, we designed a function that creates a dummy state and returns the number of features.

```
def get_num_features():
    dummy_state = {
        'round': 0,
        'step': 0,
        'field': np.zeros((17, 17)),
        'bombs': [],
        'explosion_map': np.zeros((17, 17)),
        'coins': [],
        'self': ("dummy", 0, True, (1, 1)),
        'others': []
    }

    return state_to_features(dummy_state).shape[0]
```

This makes it easier to debug problems.

1.2 Advanced features

by Mika Rother

1.2.1 Shortest path algorithms

Now we needed a shortest path algorithm, so the agent tries to find the nearest coin or crate. First we had to think about a data structure that can handle an *A* search* efficiently. For this we needed a node structure, so we created the following class

```
class Node:
    """
    This class is needed to perform an A* search
    """

    def __init__(self, parent=None, position=None):
        self.parent = parent
        self.position = position
        self.g = 0
        self.h = 0
        self.f = 0
```

```
12
13      def __eq__(self, other):
14          return self.position == other.position
15
16      def __lt__(self, other):
17          return self.g < other.g
```