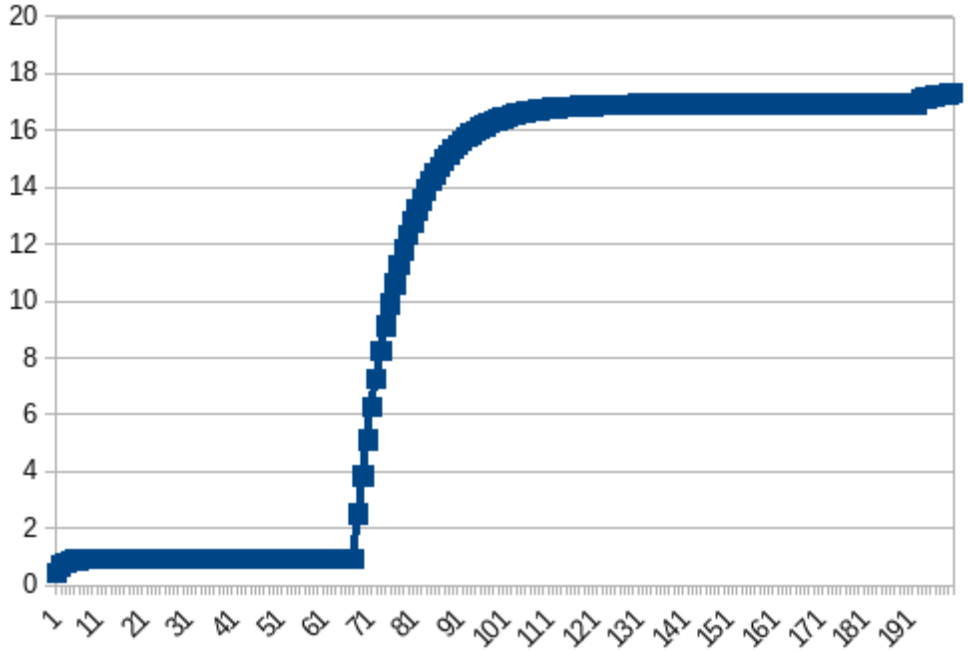
1.
    1. Q 函数的策略评价算法，使用公式 $q_{t+1}(s,a)=R(s,a)+\gamma\sum_{\acute{s}\in S}P^{a}_{s\acute{s}}\sum_{\acute{a}\in A}\pi(\acute{a}|\acute{s})q_t(\acute{s},\acute{a})$ ，对当前策略下的 Q 函数进行迭代。
    2. Q 函数的策略迭代算法，即在上一步收敛之后，改进策略为 $\acute{\pi}=argmax_{\acute{a}}(q_{\pi}(s,a))$ ，即取迭代后求得的对 s 获得最大 q 值的行为。
    3. 即通过 1 更新一次 q 函数，随后更新一次策略。
2. ε-greedy 策略能够保持策略一定的随机性，在训练过程中更充分地探索环境，可能导致 Q 函数带有随机性，减缓收敛。greedy 策略不具有随机性，在确定性问题中能够更快收敛，在有随机的环境中可能陷入局部最优解。
3.
    1. 随机策略评价：

3.30909299	8.78938844	4.42771562	5.32246388	1.49227497
1.52168472	2.99241443	2.25023639	1.907668	0.54749892
0.05091914	0.73826716	0.6732097	0.35828252	-0.40304492
-0.97349566	-0.43539886	-0.35478583	-0.58550878	-1.18297885
-1.8576039	-1.3451347	-1.22917082	-1.42282184	-1.97508282

    2. 策略迭代结果：
        1. 收敛曲线



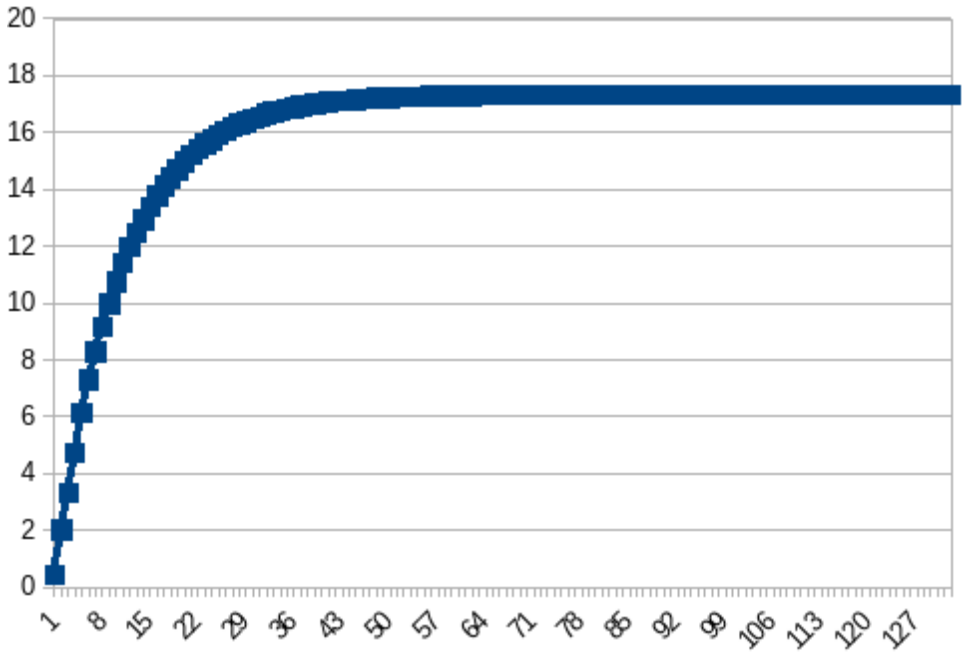
        2. 最优值函数

21.9774703176665	24.4194124671652	21.9774703176665	19.4194124671652	17.4774703176665
19.7797225034216	21.9774703176665	19.7797225034216	17.8017502530874	16.0215711909029
17.8017502530874	19.7797225034216	17.8017502530874	16.0215711909029	14.4194124671652
16.0215711909029	17.8017502530874	16.0215711909029	14.4194124671652	12.9774703176665
14.4194124671652	16.0215711909029	14.4194124671652	12.9774703176665	11.6797225034216

        3. 最优策略

3	0	2	0	2
0	0	0	2	2
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0

3. 值迭代收敛曲线



4. `inplace` 值迭代收敛曲线