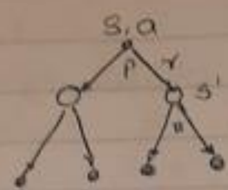
Ex - 3

1) Action Value Function.



q_π backup diagram.

=)

a) Eqⁿ for v_π in q_π & π.

$$v_\pi = \sum_{a \in A(s)} \pi(a|s)\, q_\pi(s,a)$$

Summation is overall actions a in the action space A(s) for the state

$\pi(a|s)$ is the probability of taking action a is state s under policy π.

$q_\pi(s,a)$ is expected return starting from states, taking action and thereafter following the policy.

b) Eqⁿ for q_π in terms of v_π and four - argument p.

$$q_\pi(s,a) = \sum_{s',r} p(s',r|s,a)\,[r + \gamma\, v_\pi(s')]$$

$p(s',r|s,a)$ probability of transioning s' receiving rewar in action a in s.

$v_\pi(s')$ Value of state s' under policy π.

c) Bellman eqⁿ for action values, for q_π

$$q_\pi(s,a) = \sum_{s',r} p(s',r \mid s,a)\left[r + \gamma \sum_{a'} \pi(a' \mid s')\, q_\pi(s',a')\right]$$

$p(s',r \mid s,a)$ transition probabilty function.

2) Fun with Bellman.
=)
a) Eqⁿ for v_* in terms of q_*

$$v_*(s) = \max_a q_*(s,a)$$

b) Eqⁿ for q_* in terms of v_* four-argument p.

$$q_*(s,a) = \sum_{s',r} p(s',r \mid s,a)\left(r + \gamma v_*(s')\right)$$

c) Eqⁿ for π_* in term of q_*

$$\pi_*(s) = \operatorname{argmax}_a q_*(s,a)$$

d) Eqⁿ for π_* in terms of v_* & four-argument p.

$$\pi_*(s) = \operatorname{argmax}_a \sum_{s',r} p(s',r \mid s,a) + (r + \gamma v_*($$

e) Bellman eq^n for four value functions (v_π, v_*, q_π, q_*) in terms of three argument function p

$$p(s' \mid s,a) = \sum_{r \in R} p(s', r \mid s,a) \quad \text{— (3.4)}$$

$$r(s,a) = \sum_{s',r} p(s', r \mid s,a)\, r \quad \text{— (3.5)}$$

$$\therefore\ v_\pi(s) = \sum_a \pi(a \mid s)\left[r(s,a) + \sum_{s'} p(s' \mid s,a) \times \gamma\, v_\pi(s')\right]$$

$$\therefore\ v_*(s) = \max_a \left[r(s,a) + \sum_{s'} p(s' \mid s,a) \times \gamma\, v_*(s')\right]$$

$$\therefore\ q_\pi(s,a) = r(s,a) + \sum_{s'} p(s' \mid s,a) \times \gamma \sum_{a'} \pi(a' \mid s)\, q(s',a')$$

$$\therefore\ q_*(s,a) = r(s,a) + \sum_{s'} p(s' \mid s,a) \max_a q_*(s',a)$$

3) Fixing policy iteration.

a) While fixing the bug in policy iteration, we can add a termination condition so that the policy remains unchanged for certain no. of iteration. If the policy doesn't change the predetermined no. of iteration, the algorithm terminates.

Changing the judgement condition in sudo code so that policy to be optimized.

∴ Policy - Stable $\leftarrow$ true

For each $s \in S$:

$$\text{old action} \leftarrow \pi(s)$$

$$\pi(s) \leftarrow \arg\max \sum_{s',r} p(s',r|s,a)\,[r+\gamma V(s')]$$

$$\boxed{\begin{array}{l} \text{If } q_{\text{old action}}(s',a) \neq q_{*}(s,a) \\ \text{then policy-stable} \leftarrow \text{false} \end{array}}$$

∴ The policy then stop & return $V \approx v_*$ & $\pi \approx \pi_*$ else go to 2.

b) $\cancel{v_{k+1}(s) = \max_a \sum_{s',r} p(s',r|s,a)\,[r+\gamma v_k(s')]}$

No, anologous bug in value iteration as it always selects the best action in each state based on current value funⁿ.

The policy is optimal as long as the result is optimal.

4) Policy iteration for action values:

a) 1) Initialization:

$Q(s,a) \in \mathbb{R}$ & $\pi(s) \in A(s)$ arbitarily for all $s \in S$

2) Policy Evaluation:

Loop : $\Delta \leftarrow 0$

Loop for each $s \in S$.

$a \leftarrow \pi(s)$

$q \leftarrow Q(s,a)$

$$Q(s,a) \leftarrow \sum_{s',r} p(s' r \mid s,a) \left[r + \gamma \sum_{a'} \pi(a'/s') Q(s',a') \right]$$

$\Delta \leftarrow \max(\Delta . \mid q - Q(s-a) \mid)$ [untill we get the smallest positive no.]

$\Delta < \theta$

3) Policy Improvement:

Policy - Stable $\leftarrow$ True

For each $s \in S$:

old action $\leftarrow \pi(s)$

$\pi(s) \leftarrow \underset{a}{\text{argmax}}\ q(s,a)$

If $\sum_{s',r} p(s', r \mid s, \pi(s)) [r + \gamma \sum_{a'} \pi(a'|s')\, q(a'|s')$

$\neq$

$\sum_{s',r} p(s', r) s$, old action $[r + \gamma \sum_{a'} \pi(a'|s') \times q(a';$

$\therefore$ Policy - stable $\leftarrow$ False.

& If policy - stable, then stop & return $Q \approx q_*$ & $\pi \approx \pi_*$ else we can do (2)

b) Value iteration update for all action values

$q_{k+1}(s+a)$ is $q_{(k+1)}(s,a) = \sum_{s',r} (s', r \mid s,a)$

$\theta > 0$

$[r + \gamma \max_{a'} \times q_K(s'a)]$

Initilize

$Q(s,a)$ for all $s \in S^*$, arbitiary except

the Q terminal

loop: $\Delta \leftarrow 0$

loop for each $s \in S$ $a \leftarrow \arg\max_a q(s,a)$

$q \leftarrow Q(s,a)$

$$Q(s,a) \leftarrow \sum_{s,r} p(s', r \mid s,a) \left[r + \gamma \max_a q(s',a')\right]$$

$$\Delta \leftarrow \max(\Delta \mid q - Q(s,a) \mid)$$

until $\Delta < \theta$

output a deterministic policy $\pi \approx \pi_*$ such that

$$\pi(s) : \arg\max_a \sum_{s',r} p(s', r \mid s,a) [r + \gamma q(s', \pi(s))$$

$\therefore$

$$q_{K+1}(s,a) \leftarrow \sum_{s,r} p(s', r \mid s,a) \left[r + \gamma \max_{a'} q_K(s',a')\right.$$

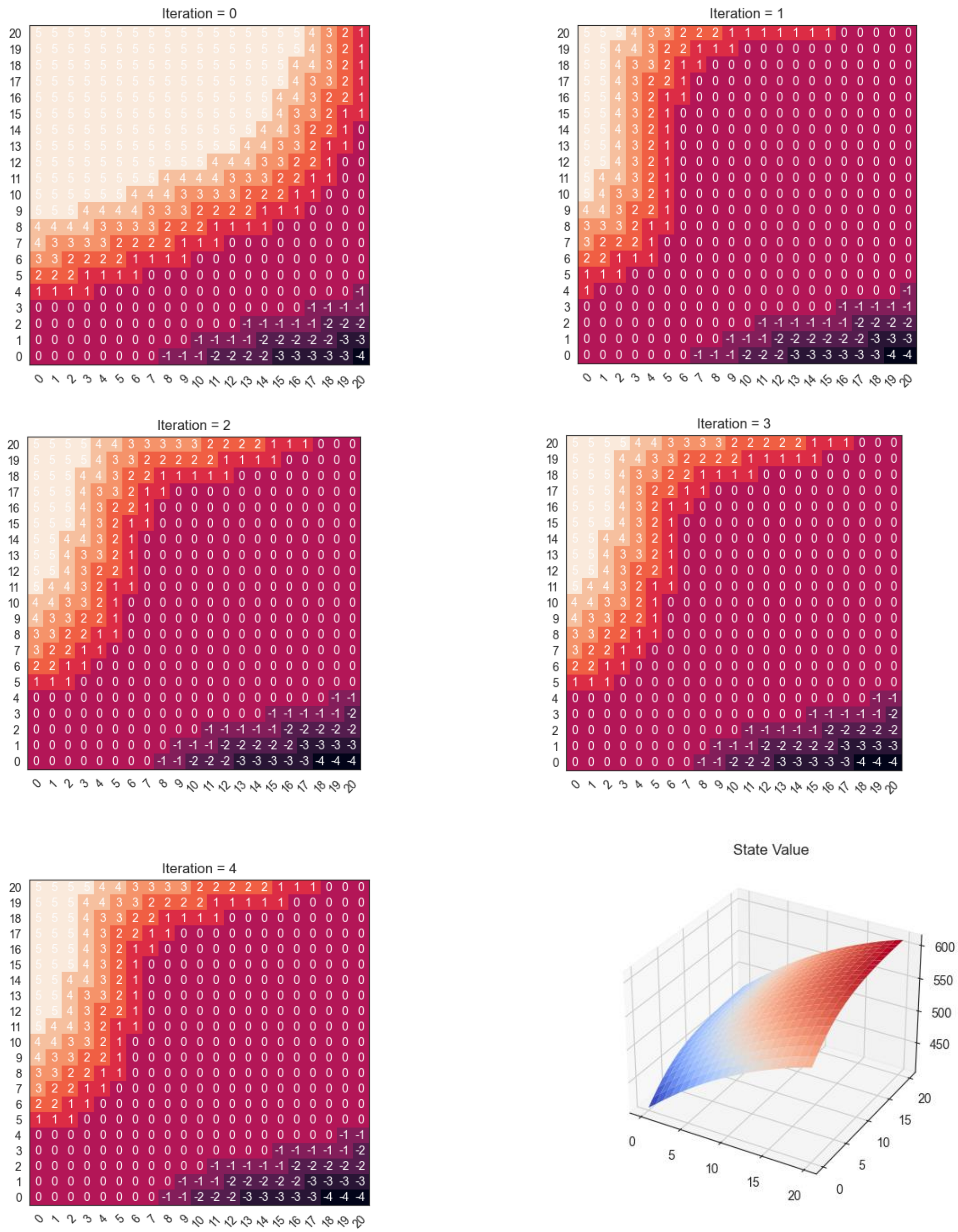
5)

a), b), c)

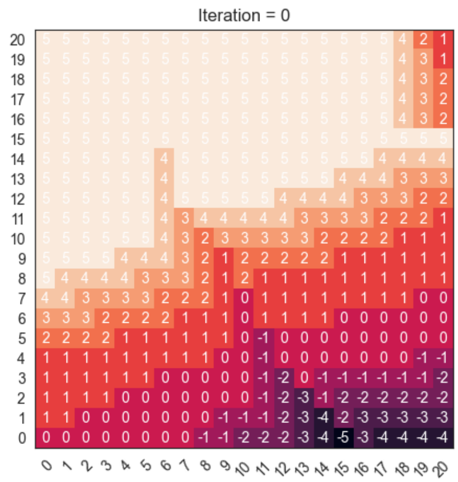
```
Answer5a V(s) -
[[ 3.31359559  8.79292942  4.43113177  5.32556099  1.4955287 ]
 [ 1.52582318  2.99591435  2.2534199   1.91064941  0.55045095]
 [ 0.05486787  0.74165922  0.67626363  0.36114423 -0.40025498]
 [-0.96965064 -0.43208514 -0.35180898 -0.58272448 -1.18027658]
 [-1.85380443 -1.34185832 -1.22622928 -1.42007309 -1.97241846]]
Answer5b V(s) -
[[21.9764967  24.41877948 21.97690153 19.41877948 17.47690153]
 [19.77884703 21.97690153 19.77921138 17.80129024 16.02116122]
 [17.80096232 19.77921138 17.80129024 16.02116122 14.4190451 ]
 [16.02086609 17.80129024 16.02116122 14.4190451  12.97714059]
 [14.41877948 16.02116122 14.4190451  12.97714059 11.67942653]]
Pi(s) -
[['right' 'up' 'left' 'up' 'left']
 ['up' 'up' 'up' 'left' 'left']
 ['up' 'up' 'up' 'up' 'up']
 ['up' 'up' 'up' 'up' 'up']
 ['up' 'up' 'up' 'up' 'up']]
Answer5c V(s) -
[[21.97748529 24.4194281  21.97748529 19.4194281  17.47748529]
 [19.77973676 21.97748529 19.77973676 17.80176308 16.02158677]
 [17.80176308 19.77973676 17.80176308 16.02158677 14.4194281 ]
 [16.02158677 17.80176308 16.02158677 14.4194281  12.97748529]
 [14.4194281  16.02158677 14.4194281  12.97748529 11.67973676]]
Pi(s) -
[['right' 'up' 'left' 'up' 'left']
 ['up' 'up' 'up' 'left' 'left']
 ['up' 'up' 'up' 'up' 'up']
 ['up' 'up' 'up' 'up' 'up']
 ['up' 'up' 'up' 'up' 'up']]
```

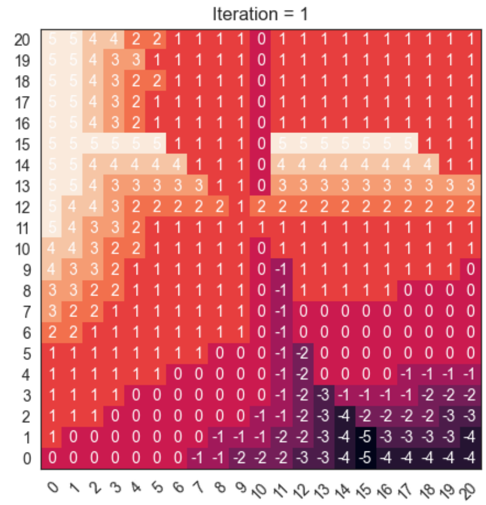
6)

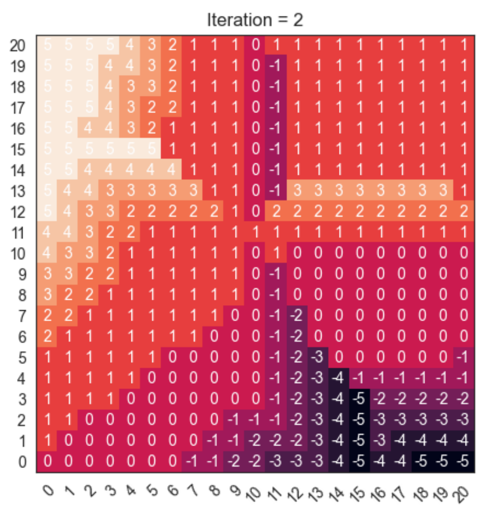
a. In these four graphs, an off-white color represents positive values, while purple denotes negative values. The color at the midpoint of each graph's color scale signifies a value of '0'.

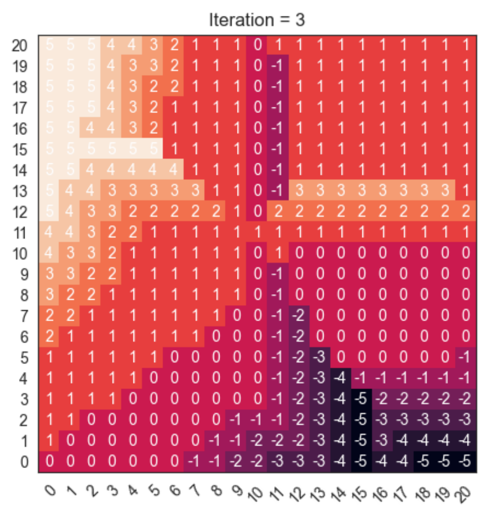


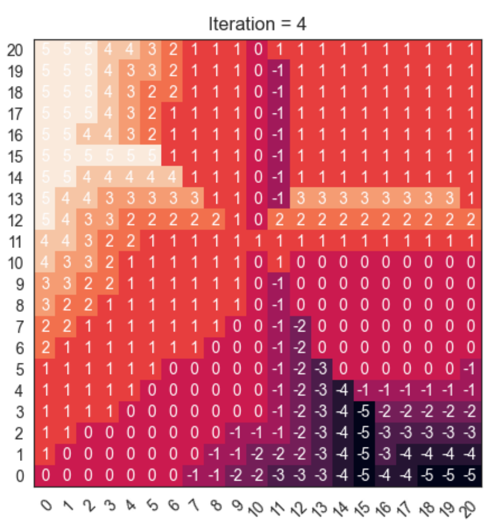
b. When modifications are implemented to the car rental system, they influence the reward structure. Specifically, if an employee helps in relocating cars, it decreases the cost associated with moving cars in one direction by a unit. This implies that the employee lives near the second location, resulting in a modified reward for transferring cars from the first location to the second.

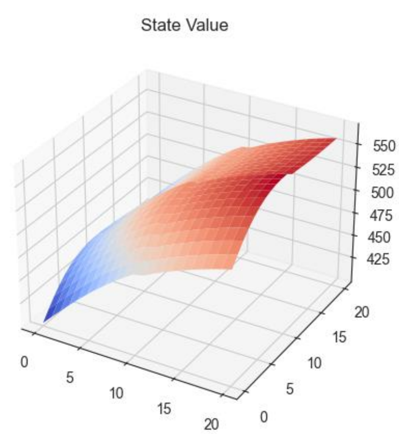












The adjustments to the car rental system's reward structure incentivize strategic car movements and inventory management: Employee Assistance: Moving a car from lot 1 to lot 2 is cheaper by one unit, encouraging movement towards lot 2 to save costs. Parking Fee: An added parking fee for more than 10 cars at any location discourages excessive inventory, promoting a strategy to keep car counts close to 10 at each location to maximize rentals while avoiding fees. Overall, these changes favor strategies that balance car availability with minimizing costs, shifting optimal operations towards maintaining moderate inventory levels at both locations to optimize profitability.

7) Extra Credit:

a) For any fun^n f and g.

$$\left| \max_a f(a) - \max_a g(a) \right| \le \max_a |f(a) - g(a)|.$$

if $\max_a f(a) - \max_a g(a) \ge 0$

$$\left| \max_a f(a) - \max_a g(a) \right| = \max_a f(a) - \max_a g(a)$$

$\max g(a) \ge g(x)$ for all x

$\max_a f(a) - \max_a g(a) \le \max_a (f(a) - g(x))$

for all x.

$$\max_a f(a) - \max_a g(a) \le \max_a (f(a) - g(a)) \quad \text{—} ①$$

if,

$a_1 = \text{argmax } f(a)$, $a_2 = \text{argmax } [f(a) - g(a)]$

then

$$f(a_1) - g(a_1) \le f(a_2) - g(a_2) \quad \text{—} ②$$

$$\left.\begin{aligned} f(a_1) - g(a_1) &= \max_a (f(a) - g(a)) \\ f(a_2) - g(a_2) &= \max_a (f(a) - g(a)) \end{aligned}\right\} ② \text{ in } ①.$$

$$\Rightarrow \max_a f(a) - g(a) \le \max_a (f(a) - g(a))$$

When,

$$\max_a f(a) - \max_a g(a) < 0$$

Substituting $f(a) \leftarrow g(a)$ & $g(a) \rightarrow f(a)$

$$\therefore \boxed{\left| \max_a f(a) - \max_a g(a) \right| \leq \max_a |f(a) - g(a)|} \quad \text{--- (3)}$$

b) Let β denote the Bellman backup operator,

Value iteration

↳ Vector form : $V_{K+1} \leftarrow \beta V_K$.

$$|\beta V_i(s) - \beta V_i(s')| = \Big| \max_a \sum_{s',r} p(s',r \mid s,a)\,[r + V_i(s')] - \max_a \sum_{s',r} p(s',r \mid s,a)\,[r + V_i(s')] \Big| \quad \text{--- (4)}.$$

Taking (3) & from a) we get,

$$\text{eq}^n\ 4 \leq \max_a \Big| \sum_{s',r} p(s',r \mid s,a)\,[r + \gamma V_K(s')] - \sum_{s',r} p(s',r \mid s,a)\,[r + \gamma V_K'(s')] \Big|$$

$$= \max_a \Big| \sum_{s',r} p(s',r \mid s,a) \times \gamma \big(V_K(s') - V_K(s') \big)$$

for all $s \in S$

for n-length V_i and V_i'.

$$\|\beta V_i - \beta V_i'\| = \max \begin{cases} |\beta V_i(S_1) - \beta V_i'(S_1)| \\ |\beta V_i(S_2) - \beta V_i'(S_2)| \\ \vdots \\ |\beta V_i(S_n) - \beta V_i'(S_n)| \end{cases} \quad \text{— (5)}$$

$$\text{eq}^n \text{ (5)} \leq \begin{cases} \max\limits_a \left| \sum\limits_{s',r} p(s',r|s,a)\, \gamma \|V_i - V_i'\| \right. \\ \max\limits_a \sum\limits_{s',r} p(s',r|s,a)\, \gamma \|V_i - V_i'\| \end{cases}$$

$$= \max \begin{cases} \gamma \|V_i - V_i'\| \\ \gamma \|V_i - V_i'\| \\ \vdots \\ \gamma \|V_i - V_i'\| \end{cases} = \boxed{\gamma \|V_K - V_K'\|}.$$

c) Banach fixed point theorem $\Rightarrow d(T(x), T(y)) \leq q\, d(x,y)$.

$\Rightarrow d(x,y) = \|x - y\|$.

$T(x) = \beta x$; β is contracting mapping

$q = \gamma$.

from (b) $\|\beta V_i - V_i'\|_\infty \leq \gamma \|V_i - V_i'\|_\infty$ — (6).

$n = \|V_{n+1} - V_n\| \leq \gamma^n \|V_1 - V_n\|$

for any $m > n$.

$$\|V_m - V_n\| \leq \|V_m - V_{m-1}\| + \|V_{m-1} - V_{m-2}\| \\ \cdots + \|V_{n+1} - V_n\|$$

$$\leq \gamma^{m-1} \| V_1 - V_0 \| + \gamma^{n-2} \| V_1 - V_0 \| \cdots \gamma^{n} \| V_1 - V_0 \| .$$

$$\leq \gamma^{n} \| V_1 - V_0 \| \sum_{k=0}^{\infty} \gamma^{k} .$$

$$\Rightarrow \| V_m - V_n \| \leq \gamma^{n} \| V_1 - V_0 \| \frac{1}{1-\gamma} < \varepsilon ,$$

$\varepsilon > 0$ is an arbitary value for large N

here seq^n.

V_i is cauchy, matching it a fixed point.

$$V^* = \lim_{n \to \infty} V_n = \lim_{n \to \infty} \beta (V_{n-1}) = \beta \left(\lim_{n \to \infty} V_{n-1} \right)$$

$$= \beta V^*$$

Coverage to fix point.

U_1, V_2 then.

$$V_1 = \beta V_1 , \quad V_2 = \beta V_2 .$$

$$\| \beta V_1 - \beta V_2 \|_\infty = \| V_1 - V_2 \|_\infty \leq \gamma \left(\| V_1 - V_2 \| \right)_\infty$$

$$V \in (0,1)$$

$\| V_1 - V_2 \| = 0 \Rightarrow$ unique fix point.

Eq. 3.9

$$\beta V_A(s) = \max_a \varepsilon \left[R_{t+1} \gamma V_A (s_{t+1}) \mid S_t = s \, A_t \right.$$

$$\Rightarrow \beta V_A(s) = \max q_{\pi_A} (s, a)$$

$$= V_* (s)$$

$\therefore$ The ~~this~~ unique fixed point is equv to Bellman.