5a) True - gradiant TD Learning Rule

$$TD(0) = w_{t+1} = w_t + \alpha\left[R_{t+1} + \gamma\hat{v}(S_t',w) - \hat{v}(S_t,w)\right]\nabla_w\hat{v}(S,w)$$

on differentiating for $\hat{v}(S',w)$ w.r.t w

$$w_{t+1} = w_t + \alpha\left[R_{t+1} + \gamma\hat{v}(S_t',w) - \hat{v}(S_t,w)\right]$$
$$\nabla_w\hat{v}(S_t,w) - \alpha\gamma\nabla_w\hat{v}(S_t',w)$$

The term $-\alpha\gamma\nabla_w\hat{v}(S_t',w)$, accounts for the change in the estimated value of next states as w changes. which makes this method a true gradiant method. As it considers the changes in w that affect the feature value estimate.

5b) The learning rule optimizes an objective func'n that includes an expectation over the next state S', making it more similar with mean squared error. of the value fun'n estimate., So theoritically

$$\Rightarrow E_\pi\left[\left(R + \gamma\hat{v}(s',w) - \hat{v}(S,w)\right)^2\right]$$

would be the objective function.

Incorporating the true could lead to more stable and accurate updates because it accurately represents the objective of minimizing the difference b/w estimated value and true value of a state across all transitions

5c) Mean Squared Bellaman Error (MSBE) objective

For the MBSE objective:

$$BE(\omega) = \sum_{s \in S} \mu(s) \left[E_\pi [R + \gamma \hat{v}(s', \omega) | S] - \hat{v}(s, \omega) \right]^2$$

To optimize this we differentiate $BE(\omega)$ w.r.t ω to find the gradient and use it in a gradient descent learning rule.

$$\therefore \nabla_\omega BE(\omega) = -2 \sum_{s \in S} \mu(s) \left(E_\pi [R + \gamma \hat{v}(s', \omega) | S] - \hat{v}(s, \omega) \right) \nabla_\omega \hat{v}(s, \omega)$$

The TD-learning rule that optimizes this objective fun would adjust the weights in the direction that minimizes the $BE(\omega)$,

$$\omega_{t+1} = \omega_t - \alpha \nabla_\omega BE(\omega)$$

5d) In Code Snippets