$MDP = \langle S, A, a, T, r, S_F, \gamma \rangle$

S :: estados
A :: Acciones
a :: Acciones legales
T :: Función de transición $S \times A \times S \to \mathbb{R}$ $T(s, a, s') = P_r[S_{t+1} = s' | S = s, A = a]$
S :: $S \to \mathbb{R}$ (s, a, s') Recompensa de ir a s' desde s haciendo a

$0 \leq \gamma \leq 1$ Factor de descuento

Retorno

$$R_t = r_t + \gamma r_{t+1} + \gamma^2 r_{t+2} \cdots$$
$$R_t = r_t + \gamma R_{t+1}$$

Encontrar una política

$\pi : S \to A$ política determinista

$$\pi(s) = a$$

Necesito comparar políticas

$$\pi_1 < \pi_2 \iff V^{\pi_1}(s) \leq V^{\pi_2}(s) \quad \forall s \in S$$

$$V^{\pi}(s) = E[R_t | S_t = s]$$

$$V^{\pi}(s) = \sum_{s' \in S} T(s, \pi(s), s') [r(s, \pi(s), s') + \gamma V^{\pi}(s')]$$

Calculo $V^{\pi}(s)$

```
def Valor_politica(π, MDP, ε):
    V(s) := random if s not in S_T else 0 for s in S
    while true:
        Δ = 0
        While true:
            for s in S/S_T
                v := V^π(s)
                V^π(s) = ΣT(s, π(s), s')[r(s, π(s), s') + γ V^π(s')
                Δ := max(Δ, |v - V^π(s)|)
            If Δ < ε break:
    return V^π
```

$$V^{\pi}(s) \leq V^{\pi^*}(s) \quad \forall s \in S \quad \forall \pi \in \Pi$$

$V^*(s)$ ← Valor óptimo

$$V^*(s) = \max_{a \in A(s)} \sum_{s' \in S} T(s,a,s')\left[r(s,a,s') + \gamma V^*(s')\right]$$

Algoritmos de programación dinámica

$$\pi_0 \rightarrow V^{\pi_0} \rightarrow \pi_1 \rightarrow V^{\pi_1} \rightarrow \pi_2 \rightarrow V^{\pi_2} \rightarrow \dots \pi^*$$

$$\pi^*(s) = \arg \max_{a \in A(s)} \sum_{s' \in S} T(s,a,s')\left[r(s,a,s') + \gamma V^*(s')\right]$$

```
def iteracion_politicas (MDP, ε):
    π(s) = random(A(s)) ∀s ∈ S/S_T
    Optima = false
    While True:
        V^π = Valor_politica (π, MDP, ε)
        Optima = true
        Para cada s en S/S_T:
            anterior = π(s)
            π(s) = arg max_{a∈A} Σ_{s'∈S} T(s,a,s') + [r(s,a,s') + γ V^π(s')]
            Si π(s) ≠ anterior
                Optima = false
        Si Optima break
    return π
```

Iteración

```
def Iteracion_valor (MDP, ε)
    V*(s) = random if S ∉ S_T else 0
    While True:
        Δ = 0
        for S in S/S_T:
            V = V*(s)
            V*(s) = max_{a∈A(s)} Σ_{s'∈S} (T(s,a,s') + [r(s,a,s') + γV*(s)]

            Δ = max (Δ, |V*(s) - v|)
        Si Δ < ε break
    for S in S/S_T:
        π*(s) = arg max_{a∈A(s)} Σ_{s'∈S} (T(s,a,s') + [r(s,a,s') + γV*(s)]

    return π
```

$O(max\_epoch * |S|^2 * |A|)$ ← polinomial