Evolucion de obligatorio IA

V1

* El algoritmo funciona y guarda los valores en la qtable considerando todas las dimensiones de la observación (velocidad angular, posición del carro, velocidad del carro, angulo del palo)
* Se comienza a partir de la versión 2 con el análisis de entrenamiento

V2

* Se utilizan valores
  + Learning rate 0.1
  + Discount 0.85
  + Cantidad de episodios 10000000000 (diez mil millones)
  + Discretización de los espacios de observación en 50 buckets (100 exceden la memoria utilizable por la ejecución)
* Se suspende el entrenamiento a los 80 millones de episodios
* Utilizando la ultima q table generada no se obtiene un reward promedio en el desempeño de una simulación de 100 episodios superior al 9
* Se intentará reducir el tamaño de la discretización de las observaciones

V3

* se disminuye la discretización a 20
* se decide realizar iteraciones de entrenamiento menos extensas que en la versión 2 para que los pasos sean mas cortos y significativos (de 1 a 10 millones de episodios)

V4

Se corren 10 millones de episodios, pero no se observa un entrenamiento adecuado del modelo ya que en promedio los rewards continúan siendo bajos.

Se incluye un código para generar métricas de rewards máximas obtenidas, mínimas y promedio en base a los episodios generados y se guardan las qtables sincronizadamente de forma que observando el grafico pueda se posible utilizar la qtable asociada al mejor desempeño.

Grafica

Gráfico, Gráfico de líneas, Histograma

Descripción generada automáticamente

Se corre una iteración de 100 episodios con la qtable guardada al inicio, a los 100 mil y a los 200 mil episodios y con la ultima qtable generada. En todos los casos se observa que el reward promedio es 10.

Impresión de pantalla luego de correr una simulación con 100 episodios. La impresión corresponde al ultimo episodio

Texto

Descripción generada automáticamente

Para la versión 4 se intentará aumentar el learning rate y disminuir a 0.5 el valor utilizado para el Discount (representando el valor asignado a la importancia de los episodios futuros frente a los actuales).

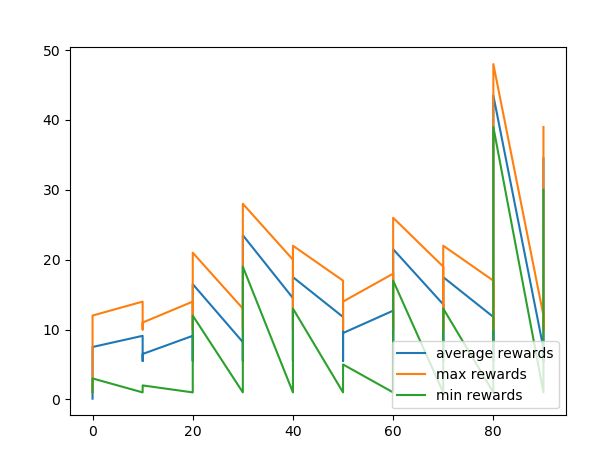
Adicionalmente se verificará que la qtable esté inicializándose, calculándose y guardándose correctamente. Se observa que ambas acciones tienen la misma “consecuencia” para nuestro modelo dado que tienen el mismo valor asociado.

V5

Debido a que se percibe que el modelo no está aprendiendo de las iteraciones realizadas, se decide comenzar nuevamente, cambiando el enfoque para una solución que solamente considere la velocidad angular y ángulo del pole para, posteriormente, incluir la velocidad y ubicación del cart en las dimensiones de la Qtable.

En el desarrollo del código, se decide implementar la discretización con KBinsDiscretizer importándolo desde sklearn.preprocessing.

Se continua con la grafica de los rewards obtenidos y corriendo una iteración de apenas 100 episodios de entrenamiento se genera la siguiente grafica:



Por lo que se utiliza la qtable guardada en la iteración 80.

Al correr una simulación de 100 episodios se obtiene

Episodio: 0 Reward obtenido: 29.0

Episodio: 1 Reward obtenido: 14.0

Episodio: 2 Reward obtenido: 31.0

Episodio: 3 Reward obtenido: 39.0

Episodio: 4 Reward obtenido: 14.0

Episodio: 5 Reward obtenido: 37.0

Episodio: 6 Reward obtenido: 45.0

Episodio: 7 Reward obtenido: 37.0

Episodio: 8 Reward obtenido: 15.0

Episodio: 9 Reward obtenido: 63.0

Episodio: 10 Reward obtenido: 42.0

Episodio: 11 Reward obtenido: 27.0

Episodio: 12 Reward obtenido: 37.0

Episodio: 13 Reward obtenido: 93.0

Episodio: 14 Reward obtenido: 27.0

Episodio: 15 Reward obtenido: 35.0

Episodio: 16 Reward obtenido: 51.0

Episodio: 17 Reward obtenido: 35.0

Episodio: 18 Reward obtenido: 71.0

Episodio: 19 Reward obtenido: 39.0

Episodio: 20 Reward obtenido: 41.0

Episodio: 21 Reward obtenido: 46.0

Episodio: 22 Reward obtenido: 27.0

Episodio: 23 Reward obtenido: 78.0

Episodio: 24 Reward obtenido: 76.0

Episodio: 25 Reward obtenido: 39.0

Episodio: 26 Reward obtenido: 17.0

Episodio: 27 Reward obtenido: 126.0

Episodio: 28 Reward obtenido: 63.0

Episodio: 29 Reward obtenido: 33.0

Episodio: 30 Reward obtenido: 15.0

Episodio: 31 Reward obtenido: 41.0

Episodio: 32 Reward obtenido: 44.0

Episodio: 33 Reward obtenido: 78.0

Episodio: 34 Reward obtenido: 33.0

Episodio: 35 Reward obtenido: 43.0

Episodio: 36 Reward obtenido: 54.0

Episodio: 37 Reward obtenido: 43.0

Episodio: 38 Reward obtenido: 45.0

Episodio: 39 Reward obtenido: 31.0

Episodio: 40 Reward obtenido: 56.0

Episodio: 41 Reward obtenido: 73.0

Episodio: 42 Reward obtenido: 50.0

Episodio: 43 Reward obtenido: 59.0

Episodio: 44 Reward obtenido: 52.0

Episodio: 45 Reward obtenido: 15.0

Episodio: 46 Reward obtenido: 82.0

Episodio: 47 Reward obtenido: 49.0

Episodio: 48 Reward obtenido: 37.0

Episodio: 49 Reward obtenido: 33.0

Episodio: 50 Reward obtenido: 27.0

Episodio: 51 Reward obtenido: 55.0

Episodio: 52 Reward obtenido: 29.0

Episodio: 53 Reward obtenido: 33.0

Episodio: 54 Reward obtenido: 46.0

Episodio: 55 Reward obtenido: 33.0

Episodio: 56 Reward obtenido: 68.0

Episodio: 57 Reward obtenido: 71.0

Episodio: 58 Reward obtenido: 37.0

Episodio: 59 Reward obtenido: 60.0

Episodio: 60 Reward obtenido: 31.0

Episodio: 61 Reward obtenido: 37.0

Episodio: 62 Reward obtenido: 50.0

Episodio: 63 Reward obtenido: 16.0

Episodio: 64 Reward obtenido: 37.0

Episodio: 65 Reward obtenido: 31.0

Episodio: 66 Reward obtenido: 53.0

Episodio: 67 Reward obtenido: 59.0

Episodio: 68 Reward obtenido: 55.0

Episodio: 69 Reward obtenido: 61.0

Episodio: 70 Reward obtenido: 33.0

Episodio: 71 Reward obtenido: 51.0

Episodio: 72 Reward obtenido: 37.0

Episodio: 73 Reward obtenido: 47.0

Episodio: 74 Reward obtenido: 31.0

Episodio: 75 Reward obtenido: 33.0

Episodio: 76 Reward obtenido: 35.0

Episodio: 77 Reward obtenido: 35.0

Episodio: 78 Reward obtenido: 40.0

Episodio: 79 Reward obtenido: 41.0

Episodio: 80 Reward obtenido: 69.0

Episodio: 81 Reward obtenido: 29.0

Episodio: 82 Reward obtenido: 31.0

Episodio: 83 Reward obtenido: 60.0

Episodio: 84 Reward obtenido: 12.0

Episodio: 85 Reward obtenido: 41.0

Episodio: 86 Reward obtenido: 19.0

Episodio: 87 Reward obtenido: 27.0

Episodio: 88 Reward obtenido: 29.0

Episodio: 89 Reward obtenido: 45.0

Episodio: 90 Reward obtenido: 33.0

Episodio: 91 Reward obtenido: 17.0

Episodio: 92 Reward obtenido: 33.0

Episodio: 93 Reward obtenido: 118.0

Episodio: 94 Reward obtenido: 26.0

Episodio: 95 Reward obtenido: 37.0

Episodio: 96 Reward obtenido: 37.0

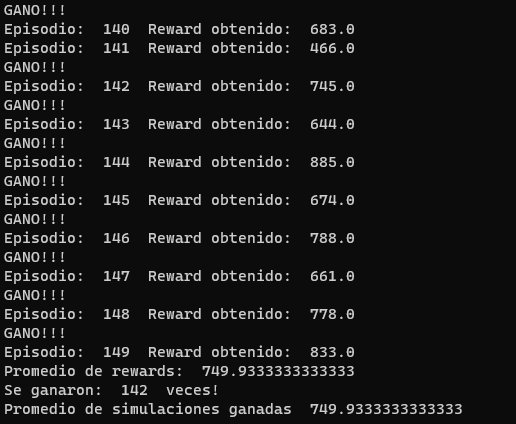
Episodio: 97 Reward obtenido: 39.0

Episodio: 98 Reward obtenido: 35.0

Episodio: 99 Reward obtenido: 56.0

Los resultados de este enfoque son mucho mas alentadores por lo que se procede a entrenar 2 millones de episodios para posteriormente verificar su desempeño.

Se observa que el entrenamiento arroja un episodio que es truncado al superar la cantidad de reward total obtenida superior a lo solicitado por el ambiente (500). Esto sucede a los 154 episodios de entrenamiento, por lo que se decide utilizar la Qtable generada (se había incluido en el código una condición para guardar la Qtable si el episodio de entrenamiento era truncado por el ambiente). Al correr 150 episodios de simulación con la Qtable anterior se obtuvo el siguiente resultado



\*Promedio de simulaciones ganadas es en realidad el reward promedio obtenido.

Como podemos apreciar se perdieron solamente 3 de los 150 episodios generados, esto induce a pensar que el error cometido es del 2% (lógicamente solo considerando los 150 episodios simulados).

Se observa un comportamiento particular: No se toma ninguna acción para que el carro no se exceda de los limites permitidos. Esto se deriva de omitir la posición del carro en las dimensiones de observación utilizada.

A continuación se adjunta la grafica del entrenamiento realizado con 2000 iteraciones:

Se realiza también un entrenamiento de 2 millones de iteraciones para verificar si el desempeño mejora con mas entrenamiento, grafica adjunta

Como se puede observar en la grafica no existe una mejora de las 2 mil a los 2 millones de iteraciones. Se procederá a entrenar el modelo con un numero de iteraciones superior (20 millones) para verificar la existencia de posibles valores locales en los que el algoritmo se pueda haber estancado, y para maximizar la ocurrencia de las observaciones de los episodios en el espacio muestral.

La versión 5 se considera una versión entregable de este desafío, pero se intentará evolucionar la entrega a una versión 6 que considere la velocidad del carro y su posición con el objetivo de no permitir que se trunque el episodio por la posición. Adicionalmente, es un desafío interesante conseguir simular solamente un episodio que sea truncado únicamente por un tiempo máximo a ser determinado.

V 5.1 Modificacion de la discretización

Se intenta ajustar la discretización para un refinamiento mas fino con 24 buckets en cada dimensión observada (velocidad angular y angulo del pole).

Al correr 2000 iteraciones no se aprecia una mejora del rendimiento

Grafica de 24 buckets por dimensión

Con un refinamiento de 10 buckets por dimensión los resultados parecen ser un poco menos volátiles

Grafica de 10 buckets por dimensión

Otros intentos fueron realizados con menor cantidad de buckets y no se observa una mejora significativa en cuanto al refinamiento, se opta por mantener los inicialmente utilizados y se procede a integrar las dimensiones de observación relacionadas al cart.

V 6 integracion de las dimensiones del cart

Se integran las dimensiones del cart y con apenas 2 mil episodios de entrenamiento no se puede distinguir una mejora en el desempeño comparándolo contra la versión 5.

Grafica de V6 con dimensión 10 para el cart

Se intenta con un refinamiento mas grueso, con 6 buckets por dimensión del cart. En esta grafica se observa un intento de continuar apendiendo en un desempeño cuya grafica del reward parece tender a subir.

Grafica de 2000 episodios con buckets en 6 por dimensión

Para verificar el comportamiento del desempeño se correrán 2 millones de episodios para comparar la gráfica correspondiente a la versión 5.

Grafica de 2 millones de episodios con refinamiento de dimensiones de cart en 6 buckets

Comparar las graficas y deducir si la versión 6 con 6 buckets por dimensión del cart es mejor que los resultados obtenidos para la versión 5 excluyendo las dimensiones del cart

Adicionalmente correr una simulación de juego con 150 episodios con la mejor qtable producida por el algoritmo considerando las dimensiones de observación del cart con los 6 buckets

Resultados de la simulación de 150 episodios con la versión 5 utilizando la qtable obtenida en el primer entrenamiento que genera una victoria (en este caso fue producida a los 139 episodios)

Texto

Descripción generada automáticamente

Resultado de 150 episodios simulados con la qtable producida en V5 a los 101 episodios

Texto

Descripción generada automáticamente

Resultado de simulación con la mejor qtable de V6 producida a los XXXXX episodios de entrenamiento

Poniendo a prueba los mejores entrenamientos obtenidos:

|  |  |
| --- | --- |
| 101-qtable.npy | Texto  Descripción generada automáticamente con confianza media |
| 128-qtable.npy |  |
| 137-qtable.npy |  |
| 154-qtable.npy |  |
| 157-qtable.npy |  |

Simulación absurda de 10000000 jugados con la 101-qtable.npy, resultado obtenido: