



Centro académico de Limón
IC-3002 Análisis de Algoritmos G-60

Proyecto 2: Blackjack

Profesor:

Joss Rayn Pecou Johnson

Hecho por:

Oscar Roni Ordoñez - 2023150387

Ricardo Arce Aguilar - 2023215990

2024

Semestre 2

Instrucciones de uso

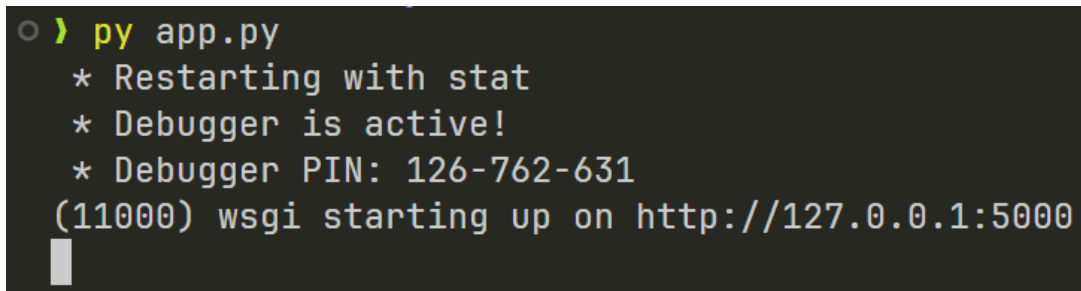
Ejecución de la aplicación

Las instrucciones de instalación y ejecución se encuentran en el siguiente repositorio de GitHub:

<https://github.com/okRoni/proyecto-2-aa.git>

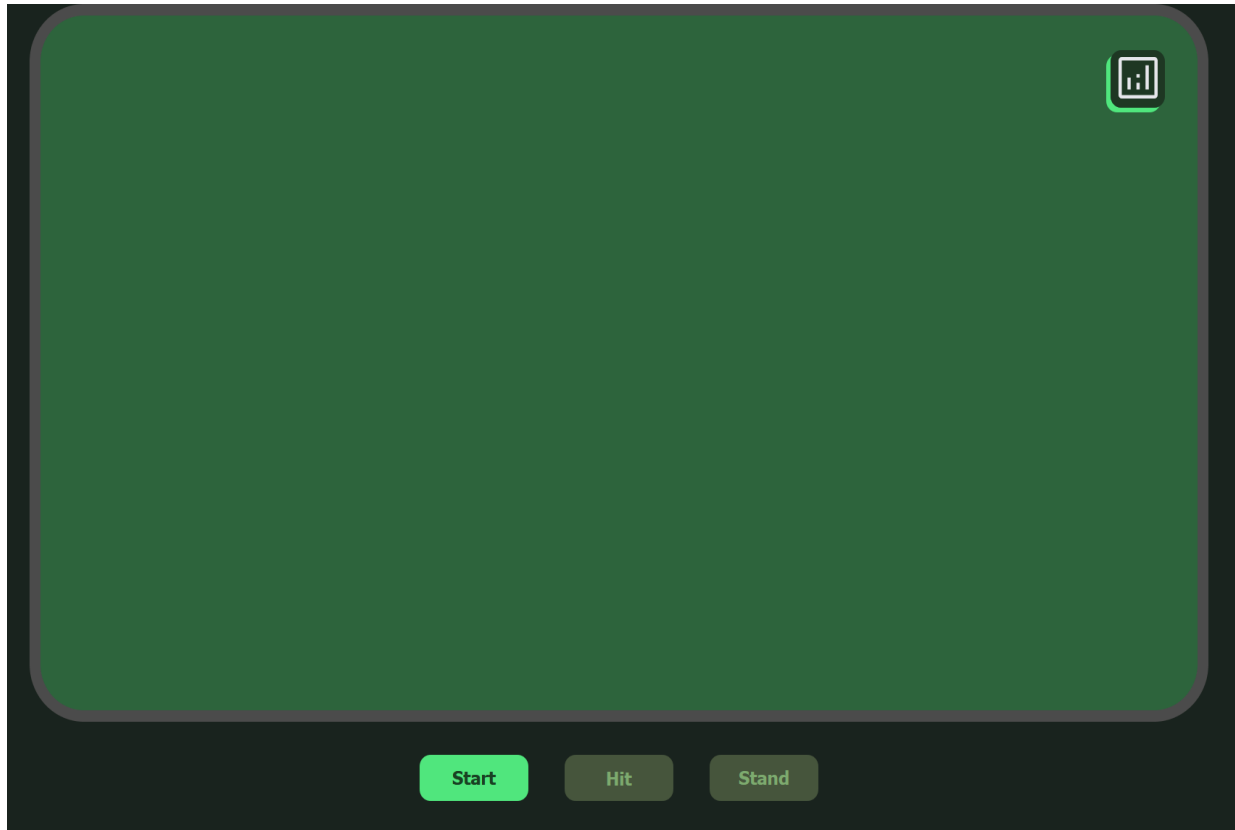
Iniciar partida

Al ejecutar la aplicación con el comando `py app.py` aparecerá en la terminal un mensaje similar al siguiente:



```
o > py app.py
  * Restarting with stat
  * Debugger is active!
  * Debugger PIN: 126-762-631
(11000) wsgi starting up on http://127.0.0.1:5000
█
```

Ahora abra su navegador web preferido e ingrese la URL que se especifica en el mensaje, <https://127.0.0.1:5000> en este caso. La URL lo llevará a la siguiente ventana:



Para iniciar el juego accione el botón “Start”. Empezará la partida y la ventana tendrá el siguiente aspecto:



Espere a que el crupier y las IAs realicen sus movimientos. Cuando terminen, se le dará la posibilidad de pedir con el botón “Hit” o plantarse con el botón “Stand”. Se le indicará además la probabilidad de no pasarse en caso de pedir.

Cuando la partida termine la aplicación mostrará el estado final de cada jugador, pudiendo ser “Busted” (se pasó), “Winner” (ganó) o “Loser” (perdió). El aspecto será el siguiente:



Si en algún momento quiere apreciar el bonito diseño de las cartas, puede ubicar su mouse encima de alguna de ellas y se pasará al frente. Por ejemplo, ubicando el mouse encima de la segunda carta propia, se traerá al frente y se verá de la siguiente manera:



Ver estadísticas

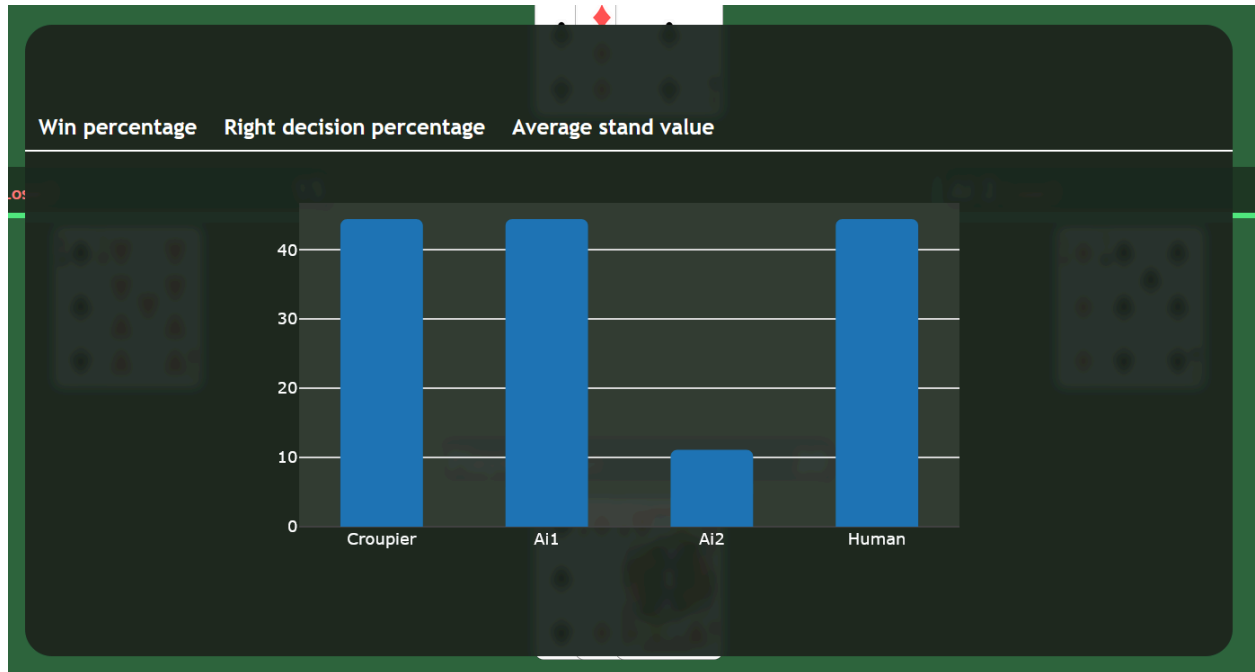
Una vez haya terminado su partida, opcionalmente puede accionar el botón de estadísticas ubicado en el extremo superior derecho.



Podrá acceder a las siguientes estadísticas:

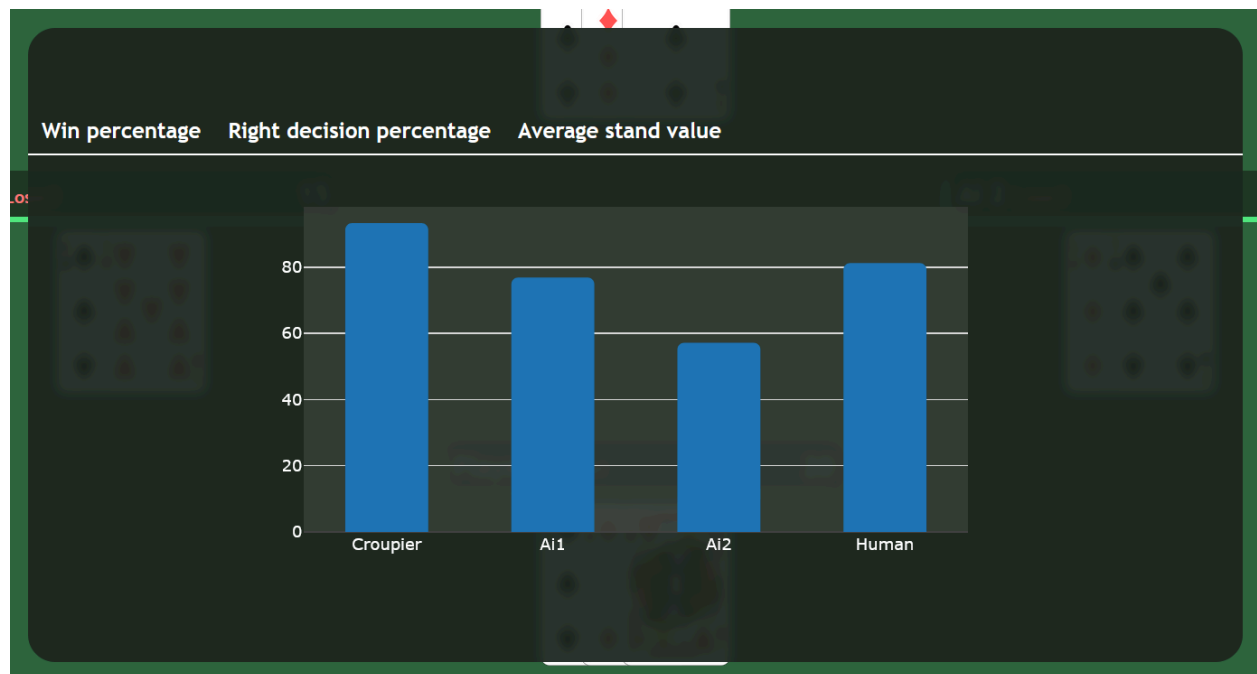
Win percentage

Aquí se muestra cuánto porcentaje de partidas ha ganado cada jugador. Se calcula dividiendo el total de partidas ganadas entre el total de partidas jugadas por cada jugador. Su aspecto es el siguiente:



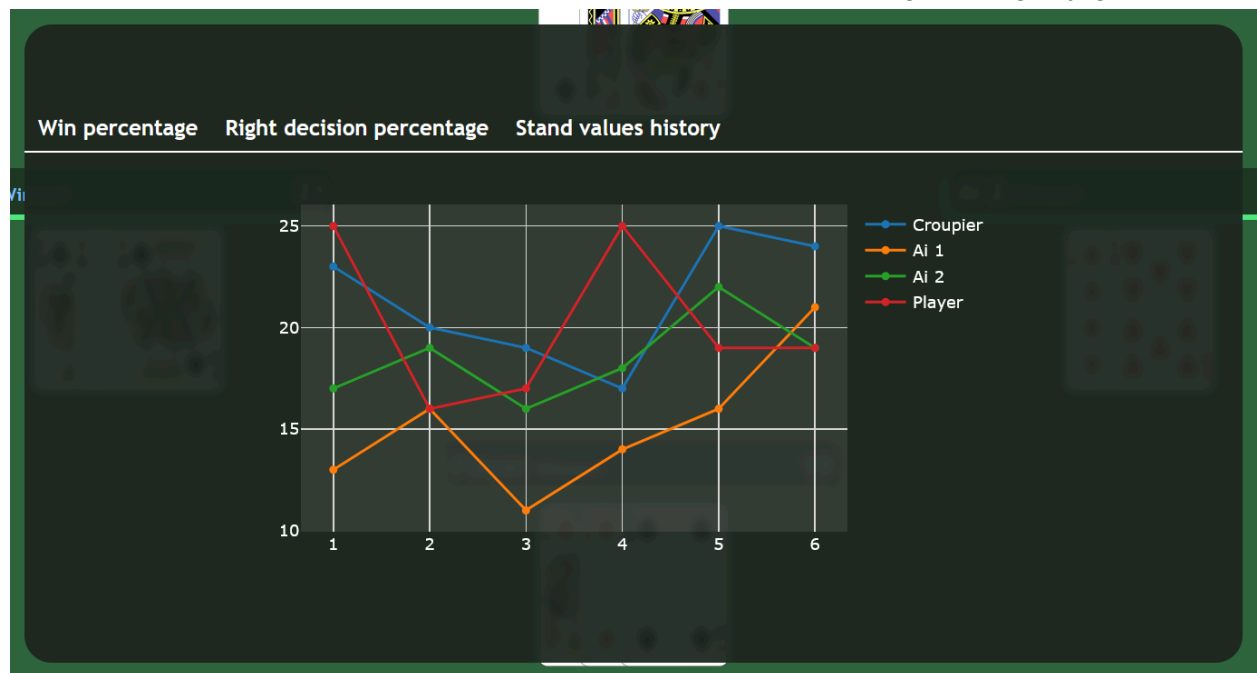
Right decision percentage

Aquí se muestra cuánto porcentaje de decisiones correctas ha tomado cada jugador. Se calcula dividiendo el total de decisiones (movimientos) beneficiosos entre el total de decisiones tomadas por cada jugador. Su aspecto es el siguiente:



Stand values history

Aquí se muestra el valor de la mano con el cual se ha plantado cada jugador a lo largo de las partidas. De ahí puede apreciar cualquier cambio en la estrategia de algún jugador.



Decisiones de diseño

Algoritmo basado en Q-Learning

Según se investigó, el algoritmo Q-Learning es un tipo de aprendizaje automático reforzado, el cual se basa en almacenar una tabla de valores llamada “Tabla Q” en la que sus filas representan estados y sus columnas representan acciones. Con cada acción que el algoritmo toma, se actualiza el valor de la Tabla Q correspondiente al estado actual y la acción tomada en base a una “Recompensa”, un valor obtenido de analizar el beneficio obtenido de realizar la acción indicada en el estado indicado. Más tarde se usan los valores de la Tabla Q para determinar cuál es la mejor acción posible a realizar dado el estado actual.

El algoritmo inicia con una fase de entrenamiento en la cual se empiezan realizando acciones mayormente aleatorias con las cuales el algoritmo puede “explorar” su entorno. Después de cierto número de acciones tomadas o de llegara cierta estabilidad en los valores de la Tabla Q, se inicia con la etapa de predicción, en la que se usan los valores de la Tabla Q para tomar las mejores decisiones en cada estado en el que se puede encontrar el algoritmo en su entorno.

Para realizar el paso de actualizar cada valor de la Tabla Q en base a cierta recompensa, se usa la siguiente fórmula:

$$Q(s, a) = Q(s, a) + t(r + d \cdot m - Q(s, a))$$

Las variables involucradas corresponden a:

- $Q(s, a)$: Valor de la Tabla Q en la posición correspondiente al estado s y acción a .
- t : Tasa de aprendizaje. Establece qué tan rápido cambiarán los valores de la tabla en base a las recompensas recibidas.
- r : Recompensa (o penalización) por realizar la acción a en el estado s .
- d : Factor de descuento. Establece qué tan rápido cambiarán los valores de la tabla en base al mejor estado siguiente posible.
- m : Valor Q del mejor estado siguiente posible.

Además de los parámetros anteriores, se establece la constante p , que corresponde a la probabilidad de exploración. Es decir, a la probabilidad de ignorar la mejor acción posible conocida y tomar en su lugar una acción aleatoria.