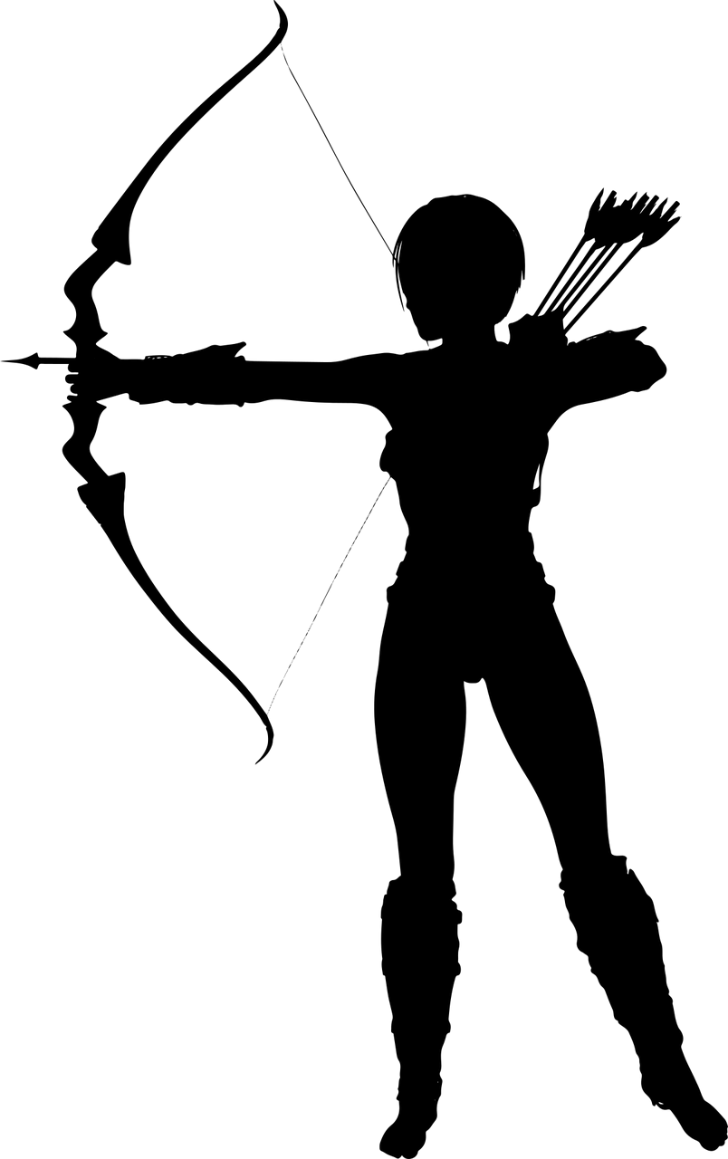
El Favor de las guerreras



Contenido

[1. Uso e instalación 2](#_Toc72781244)

[1.1 Despliegue del entorno 2](#_Toc72781245)

[1.2 Ejecución y configuración 2](#_Toc72781246)

[2. Interfaces y alcance 3](#_Toc72781247)

[2.1 Contexto 3](#_Toc72781248)

[2.1.1 Introducción y objetivos 3](#_Toc72781249)

[2.1.2 Reglas del juego 3](#_Toc72781250)

[3. Desarrollo 6](#_Toc72781251)

[3.1 Estrategia de solución 6](#_Toc72781252)

[3.1.1 Disposición del tablero y transformación de este a datos útiles 6](#_Toc72781253)

# Uso e instalación

## Despliegue del entorno

Descargar en instalar [Anaconda](https://www.anaconda.com/products/individual)

Una vez instalado Anaconda, desde la aplicación de promt de anaconda Anaconda Prompt crear un entorno nuevo con el comando:

*conda create --name <nombre del entorno>*

Para activar el entorno creado ejecutar el comando:

*conda activate <nombre del entorno>*

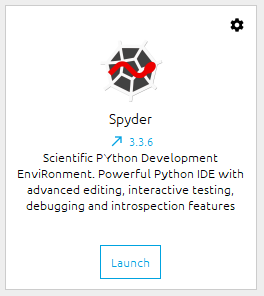
Abrir el navegador de Anaconda Anaconda Navigator, en Aplications on seleccionar el entorno creado. A continuación, abrir la sección Environments y buscar e instalar:

* Python 3.7.9
* Spyder 3.3.6
* Tensorflow 2.3.0
* ConfigParser 5.0.2

Una vez instaladas las librerías, abrir Spyder desde Home del navegador de Anaconda

## Ejecución y configuración

Para la ejecución de la aplicación abrir el programa Spyder instalado dentro de Anaconda

[](https://github.com/samuelmorenov/El-Favor-de-las-Guerreras/blob/main/doc/images/spyder.png)

Desde Spyder abrir el archivo *main.py* que está en *\src\main\python* y pulsar F5 para ejecutar.

Para cambiar la configuración, abrir el archivo *param.properties* que está en \src\main\resources. Hay 3 campos configurables:

* MODO: Para elegir entre **Generar datos** (1), **Entrenar a la red neuronal** (2) o **Jugar** (3)
* NUM\_SIMULACIONES: En caso de haber elegido el modo de generar datos, este número indica el número de partidas que se van a simular
* DIFICULTAD: Para elegir entre **Bot de acciones aleatorias** (1) o **Red neuronal entrenada** (2)

# Interfaces y alcance

## Contexto

### Introducción y objetivos

El trabajo consiste en una versión en digital del juego de mesa Hanamikoji, en el que el jugador se enfrenta a una inteligencia artificial entrenada usando una red neuronal que aprende a jugar usando aprendizaje profundo. Para ello se ha implementado la lógica del juego en Python y se le suministrará a la librería de Python TensorFlow, en la que se programará la red neuronal. De manera que llegue a tener un nivel suficiente como para llegar a ser un reto para el jugador.

### Reglas del juego

#### Preparación

Coloca 7 Cartas de Geisha en una fila, en el siguiente orden de izquierda a derecha, entre los jugadores.

Coloca 1 Marcador de Victoria en el centro de cada Carta de Geisha.

Apila las Cartas de Objeto boca abajo en un mazo y colócalo a un lado.

Cada jugador coge 4 Marcadores de Acción del mismo color con el lado coloreado hacia arriba y los coloca frente a él.

El jugador más joven es el jugador inicial.

#### Secuencia de Juego

El juego se desarrolla durante una o varias rondas. Cada ronda consiste en 3 fases en el siguiente orden:

Fase 1: Reparto

Fase 2: Acción

Fase 3: Puntuar y Actualizar

Si cualquier jugador consigue el objetivo de victoria en la fase de Puntuar, el juego finaliza inmediatamente. Si ningún jugador consigue el objetivo de victoria, el juego prosigue a la siguiente ronda. El juego continuará hasta que alguno de los jugadores gane.

##### Fase 1: Reparto

El jugador inicial baraja las 21 Cartas de Objeto y las apila boca abajo, y al azar retira 1 carta del mazo y la devuelve a la caja del juego sin mirarla. Esta carta no se usará en esta ronda. Ningún jugador tiene permitido revisarla.

Reparte a cada jugador 6 Cartas de Objeto para conformar su mano, que se mantendrá oculta.

Apila el resto de Cartas de Objeto boca abajo como “Mazo de Objetos” y colócalo a un lado de la fila de Cartas de Geisha.

##### Fase 2: Acción

Empezando por el jugador inicial, los jugadores se alternan en turnos (Jugador A -> Jugador B -> Jugador A -> Jugador B -> Etcétera) hasta que ambos jugadores hayan realizado 4 turnos.

En tu turno, debes coger una carta del Mazo de Objetos y realizar una acción.

Cuando realizas una acción, escoges 1 de tus Marcadores de Acción del lado colorado y realizas la correspondiente acción. Tras resolverla, coloca boca abajo el marcador. No puedes usar marcadores boca abajo (los marcadores de acción de cada jugador se podrán usar una sola vez por ronda.

Hay 4 acciones en Hanamikoji:

#### 1 Secreto

Escoge una carta de tu mano y colócala boca abajo debajo del Marcador de Acción usado (Secreto). Esta carta se revelará en la Fase de Puntuar y se puntuará.

Puedes revisar esta carta en cualquier momento.

#### 2 Renuncia

Escoge 2 cartas de tu mano y colócalas boca abajo frente a ti, debajo del Marcador de Acción usado (Renuncia). Estas cartas no puntuarán en esta ronda.

Puedes revisar estas cartas en cualquier momento.

#### 3 Regalo

Escoge 3 cartas de tu mano y colócalas boca arriba frente a ti.

Tu oponente escoge 1 de esas cartas y la coloca en su lado junto a la correspondiente Geisha. Tú colocas las otras 2 cartas en tu lado junto a la(s) correspondiente(s) Geisha(s). Estas cartas puntuarán.

#### 4 Competición

Escoge 4 cartas de tu mano y colócalas boca arriba frente a ti. Divídelas en dos grupos, cada uno con 2 cartas.

Tu oponente escoge 1 grupo y coloca las 2 cartas en su lado junto a la(s) correspondiente(s) Geisha(s). Tú colocas las otras 2 cartas en tu lado junto a la(s) correspondiente(s) Geisha(s). Estas cartas puntuarán.

##### Fase 3: Puntuar y Actualizar

Después de que ambos jugadores hayan realizado 4 acciones, el juego prosigue a la Fase 3.

Los jugadores revelan la carta debajo del Marcador de Acción (Secreto) y la colocan en su lado junto a la correspondiente Geisha. Compara el número de Cartas de Objeto en ambos lados de cada Geisha.

- Un lado tiene más cartas que el otro: El lado con más Cartas de Objeto gana el Favor de la Geisha. Mueve el Marcador de Victoria al lado vencedor.

- Ambos lados empatan o no hay cartas: No se mueve el Marcador de Victoria.

Tras puntuar, los jugadores calculan el número de Geishas cuyo Favor han ganado y suman sus Puntos de Carisma. Si algún jugador alcanza el objetivo de victoria, el juego finaliza inmediatamente. (Ver Finalización del Juego).

Si ningún jugador consigue el objetivo de victoria, se procede a Actualizar:

- Recoge TODAS las Cartas de Objeto de la mesa y de la caja, apílalas boca abajo en un mazo y colócalo a un lado.

- Los Marcadores de Victoria permanecerán en su lugar. Nota: No los devuelvas al centro de cada Carta de Geisha.

- Los jugadores colocan boca arriba sus Marcadores de Acción.

- El segundo jugador se convierte en el jugador inicial.

- La siguiente ronda está lista para empezar.

#### Finalización del Juego

Si algún jugador gana el Favor de 4 Geishas u 11 (o más) Puntos de Carisma, el juego finaliza.

Si un único jugador alcanza el objetivo de victoria, es el vencedor.

Si un jugador gana el Favor de 4 Geishas y el otro gana 11 (o más) Puntos de Carisma, éste último vence.

# Desarrollo

## Estrategia de solución

### Disposición del tablero y transformación de este a datos útiles

#### Elementos del tablero

Como se describe en las reglas del juego, hay 7 cartas centrales que en este caso llamaremos guerreras con diferentes valores.



Estos valores vienen dados por el número de cartas de favor que hay para cada guerrera en el mazo de cartas.

##### Favor de las guerreras

Las guerreras tienen un marcador de favor, que empieza estando imparcial y al final de cada turno cambiará dependiendo del favor que haya conseguido cada jugador. Por tanto, tendremos que guardar un array de 7 campos, uno para cada guerrera, con valores de 0 si ningún jugador tiene su favor, 1 si el jugador 1 tiene su favor y 2 si lo tiene el jugador 2.

##### Cartas en la mano

Cada jugador tendrá cartas en su mano, llegando a tener como máximo 7 cartas. Por lo que habrá que guardar los datos de las cartas de cada jugador, para ello se utilizará un array para cada jugador de 7 campos con valor 0 si es un hueco y valores del 1 al 7 de manera que el 1 corresponde a una carta de favor de la guerrera 1 y así sucesivamente hasta el 7.

##### Acciones realizadas

Cada jugador tendrá la opción de realizar las 4 acciones por turno, por lo que hay que guardar los datos de acciones se han realizado y cales no para cada jugador. Además, en caso de la acción 1 hay que guardar la carta que se utilizo para el secreto, y en el de la acción 2 hay que guardar las 2 cartas que se descartaron para la renuncia. Así pues, se guardarán todos estos datos en un único array para cada jugador con 5 posiciones con la siguiente configuración:

* Posición 0: guarda el valor de la carta de secreto (0 en caso de no haber sido usado)
* Posición 1 y 2 valores de las cartas de renuncia (0 en caso de no haber sido usado)
* Posición 3 guarda si se ha usado el regalo (0 si no se ha usado, 1 si se ha usado)
* Posición 4 guarda si se ha usado la competición (0 si no se ha usado, 1 si se ha usado)

##### Armas usadas

Las armas (que son las cartas que se roban del mazo de armas y se usan para subir el favor de las guerreras) que cada jugador ha utilizado. Estas se guardarán en un array para cada jugador, en el que cada campo corresponde a una de las guerreras y el valor de este campo es el numero de armas que ha usado en esa guerrera.

##### Acción de decisión

En caso de que en que un jugador haya jugado la acción 3 o la acción 4, el otro jugador deberá decidir entre las opciones que se le ofrecen. Para esto guardaremos las cartas que se ofrecen junto al tipo de acción en un array auxiliar, donde la primera posición será el valor de la acción y las siguientes las cartas a elegir. En el caso de que sea la acción 4 estarán en orden, la posición 1 y 2 corresponderán a un grupo y la posición 3 y 4 al otro grupo.

#### Matriz de información

Resumiendo el punto anterior, tenemos 2 array para las manos de los jugadores, 2 array para las armas usadas, 2 array para las acciones realizadas, 1 array para el valor del favor actual y 1 array para la acción de decisión. Aunque cada array tiene un tamaño distinto, para poder conformar una matriz con toda la información tomaremos el valor máximo de todas ellas, que es 7. De esta manera tenemos una matriz de 7x8 con toda la información de un estado concreto del tablero de juego.

##### Transformación de la matriz

A la hora de enviar esta información a un jugador esta

##### Un ejemplo de matriz

0000023 -> Cartas en la mano del jugador 1 (2 y 3)

0000000 -> Mano vacía del jugador 2

4001100 -> Acciones realizadas por el jugador 1 (ha realizado las acciones 3, 4 y 1 (usando la carta 4))

1111100 -> Acciones realizadas por el jugador 2 (ha realizado todas sus acciones)

0001213 -> Armas utilizadas por el jugador 1 (1 en la guerrera 4, 2 en la guerrera 5, ...)

0110122 -> Armas utilizadas por el jugador 2 (1 en la guerrera 2, 1 en la guerrera 3, ...)

0201210 -> Favor de las guerreras (el jugador 1 gana en las 4 y 6 y el jugador 2 en las 2 y 5)

0000000 -> Acción de decisión, al estar vacía hay que realizar una acción normal

#### Acción decidida por el jugador

Ya sea un usuario mediante la interfaz gráfica, o la red neuronal, recibirá la matriz descrita anteriormente con la información de la situación actual del tablero, con esta información el jugador realizará una acción dentro de las 4 posibles o una acción de decisión dentro de las 2 posibles. Además, todas las acciones van acompañadas de un numero de cartas entre 1 y 4 dependiendo de la acción. Por lo tanto, la acción completa se enviará como un array de 5 campos, donde el primer campo es la acción por realizar (0, 1, 2 o 3 para las acciones normales, 4 para la acción de decisión de la acción de regalo y 5 para la acción de decisión de la acción 4) y el resto de los campos las cartas seleccionadas (0 en caso de que la acción no requiera todos los campos).

Un ejemplo de acción

13200 -> El jugador realiza la acción de renuncia con las cartas 3 y 2