

Projeto

Poker simplificado

Para 27 de Julho de 2022

1 O Jogo

Vamos trabalhar neste projeto uma versão de Poker com 2 jogadores apenas. O baralho será composto por 5 cartas, arbitrariamente nomeadas:

$$A, 7, K, Q, J.$$

As cartas *não* possuem “naipe”, e estão repetidas exatamente 5 vezes no baralho, que, portanto, possui 25 cartas ao todo.

Uma partida consiste em 5 *rodadas*, todas elas semelhantes, a menos da ordem dos jogadores, que alterna a cada rodada. Uma rodada *completa* tem as seguintes etapas

1. Cada jogador recebe 5 cartas, aleatoriamente, do baralho.
2. O primeiro jogador decide se deseja trocar até 3 de suas cartas.
3. O primeiro jogador recebe suas novas cartas (e o segundo *sabe* quantas foram trocadas, mas não quais).
4. O segundo jogador também decide trocar até 3 cartas.
5. O segundo jogador também recebe suas cartas.
6. O primeiro jogador pode desistir. Neste caso, o segundo jogador ganha *dois* pontos, e encerra-se a rodada.
7. Se o primeiro jogador não desistiu, o segundo jogador pode desistir. Neste caso, o primeiro jogador ganha *três* pontos, e encerra-se a rodada.
8. Enfim, se nenhum jogador desistiu, os jogadores revelam suas cartas. Aquele que tiver a mão com mais pontos ganha a rodada e recebe os pontos correspondentes; o outro jogador não faz nenhum ponto. Em caso de empate, ambos jogadores ganham metade dos pontos.

Uma rodada *rápida* tem apenas as etapas 1 e 8. Uma rodada *estratégica* tem as etapas 1, 2, 3, 4, 5 e 8.

A “mão” de um jogador de Poker são 5 cartas que estão com um jogador. As mãos têm os seguintes valores:

Um par: 1 ponto	Uma tripla: 3 pontos	Uma quadra: 10 pontos
Dois pares: 2 pontos	Uma tripla e um par: 5 pontos	Um quina: 50 pontos

2 Preliminares

Uma parte fundamental do projeto consiste em organizar a dinâmica do jogo. Para isto, vamos escrever algumas funções auxiliares, testá-las, para em seguida consolidar todas numa partida.

2.1 Representação computacional

Adotaremos as seguintes convenções ao longo deste projeto:

- Uma carta será designada pela *string* correspondente;
- Uma mão será dada por uma lista de 5 elementos;
- Cada jogador será designado por uma *string*;
- O estado da rodada será armazenado em uma tupla, com a mão do primeiro jogador e a mão do segundo jogador.
- O estado do jogo será armazenado em uma tupla contendo: o número da rodada atual; os nomes dos jogadores; um dicionário com o placar atual.

Questão 1. Valor das mãos

- Escreva uma função que recebe uma mão e retorna o seu valor em pontos.
- Escreva uma função que recebe duas mãos e retorna uma tupla, com o número de pontos que cada jogador vai receber.
- Escreva uma função que recebe uma mão e retorna uma lista das cartas que *não* contribuem para o valor da mão.
- Escreva uma função que recebe uma mão e retorna uma lista de no máximo 3 cartas que não contribuem para o valor da mão.

Questão 2. Trocas estratégicas. Represente as cartas que não estão em uma mão por uma lista de 20 cartas. Neste exercício, vamos simular todas as possibilidades de troca. Dica: leia a documentação de `itertools.combinations`.

- Gerador de trocas.** Escreva uma função `possibilidades(mão_parcial, num_cartas, cartas_restantes)` que retorna um gerador com todas as mãos possíveis ao combinar as cartas de `mão_parcial` com o sorteio de `num_cartas` dentre as `cartas_restantes`.
- Um par.** Considere uma mão com um par apenas. Em quantas ocasiões trocar as 3 outras cartas resultará em uma mão melhor? Em quantas ocasiões nada acontece?

Note que, mesmo jogando contra outro jogador, você deve considerar as 20 cartas que não estão na sua mão, pois ignoramos quais 5 foram distribuídas para o adversário.

- Mais trocas com um par.** Continuando para apenas um par, quais são os resultados para trocar apenas 2 cartas? E para trocar apenas uma carta?
- Valor esperado.** Escreva uma função `valor_médio_troca(mão, trocas)` que retorna a *média* do valor de uma mão, ao trocar a lista de cartas em `trocas`.

3 Jogadores simples

Um jogador de poker é definido por duas funções: A primeira, `trocas(mão, trocas1=None)`, recebe uma mão (e, se for o segundo jogador, quantas o primeiro jogador trocou) e decide quais cartas trocar. A segunda, `desistir(mão_atual, mão_original, trocas_minhas, trocas_outro=None, segundo=False)`, retorna `True` se o jogador irá desistir (o parâmetro `segundo` é `True` se este for o segundo jogador).

Questão 3. Jogador Aleatório. Implemente um jogador aleatório: ele dá 50% de probabilidade para cada cartas de sua mão original ser trocada (até um máximo de 3), e desiste com 50% de probabilidade.

Suas funções devem se chamar `trocas_aleatório` e `desistir_aleatório`.

Questão 4. Jogador Humano. Implemente um jogador “humano”: ele mostra as cartas da mão, e pergunta quais serão trocadas. O jogador então dá o nome das cartas a serem trocadas, e o jogador retorna a lista correspondente.

A outra função, por sua vez, pergunta se sim ou não o jogador deseja desistir. Você pode implementar um valor padrão, ou exigir uma resposta correta ;-)

Por exemplo, uma rodada poderia ser

Seu adversário trocou 2 cartas. Você possui [A, K, 7, A, Q].

Quais deseja trocar? K7A

Seu adversário decidiu continuar.

Sua mão agora é [Q, J, J, A, Q]. Quer continuar? Sim

Suas funções devem se chamar `trocas_humano` e `desistir_humano`.

Questão 5. Jogador ganancioso. Implemente um jogador simples: ele sempre troca as cartas que não contribuem para o valor da mão. Se uma mão não tem nem um par, ele troca as 3 menores.

Para desistir, ele desiste se tem um par ou menos e é o primeiro jogador, e ele desiste se tem não tem uma tripla e é o segundo jogador.

Suas funções devem se chamar `trocas_ganancioso` e `desistir_ganancioso`.

Questão 6. Jogador cauteloso. O jogador cauteloso troca da mesma forma que o ganancioso. Ele é apenas diferente na hora de desistir: se o adversário trocou apenas uma carta, ele assume que o oponente tem ao menos uma quadra; se o adversário trocou apenas duas cartas, ele assume que o oponente tem uma tripla.

Suas funções devem se chamar `trocas_cauteloso` e `desistir_cauteloso`.

4 Jogadores de competição (opcional, mas divertida!)

Questão 7. Jogador blefe. Este jogador sempre troca no máximo uma carta, e desiste como o jogador ganancioso. Como ele se comporta contra os outros jogadores?

Questão 8. Seu jogador. Implemente (e descreva nos comentários!) uma estratégia de sua escolha. Como ele se comporta contra os outros jogadores que você implementou?