Package 'axisandallies'

January 27, 2025

Type Package

Title Axis and Allies Spring

Version 0.1.1

Description Simulates battles in the board game Axis and Allies Spring 1942, and calculates your probability of winning a battle. This speeds the game up significantly.

License MIT + file LICENSE

Encoding UTF-8

RoxygenNote 7.2.3

NeedsCompilation no

Author TJ Weaver [aut, cre]

Maintainer TJ Weaver <weaverthomasjohn@gmail.com>

Repository CRAN

Date/Publication 2025-01-27 13:10:01 UTC

Contents

buy_units *Buy Units*

Description

Calculates the cost of a purchase during the purchase units phase

Usage

```
buy_units(
  infantry = 0,
  artillery = 0,
  tanks = 0,
  fighters = 0,
  bombers = 0,
  aaguns = 0,
  complexes = 0,
  submarines = 0,
  destroyers = 0,
  carriers = 0,
  cruisers = 0,
  battleships = 0
)
```

Arguments

infantry	Number of infantry purchased, infantry cost three
artillery	Number of artillery purchased, artillery cost four
tanks	Number of tanks purchased, tanks cost five
fighters	Number of fighters purchased, fighters cost ten
bombers	Number of bombers purchased, bombers cost twelve
aaguns	Number of anti aircraft guns purchased, anti aircraft guns cost five
complexes	Number of industrial complexes purchased, industrial complexes cost fifteen
submarines	Number of submarines purchased, submarines cost six
destroyers	Number of destroyers purchased, destroyers cost eight
carriers	Number of aircraft carriers purchased, aircraft carriers cost fourteen
cruisers	Number of cruisers purchased, cruisers cost twelve
battleships	Number of battleships purchased, battleships cost twenty

Value

Numerical cost of purchase

Examples

```
buy_units(infantry = 3, artillery = 1, tanks = 1, complexes = 1, submarines = 2)
```

info_units *Unit Information*

Description

Gives basic information about the units in axis and allies

Usage

```
info_units(unit, write_to_console = TRUE)
```

Arguments

unit	A unit in axis and allies spring 1942 in all lowercase letters
write_to_console	If true, writes the output to the console, if false, returns as a vector

Value

Describes the unit's attack, defense, movement, and cost, and other details in several lines of text

Examples

```
info_units("artillery")
```

land_battle *Run a Land Battle*

Description

Simulates one land battle for given attacking and defending units

Usage

```
land_battle(
  offense_infantry = 0,
  offense_artillery = 0,
  offense_tanks = 0,
  offense_fighters = 0,
  offense_bombers = 0,
  defense_infantry = 0,
  defense_artillery = 0,
  defense_tanks = 0,
  defense_fighters = 0,
  defense_bombers = 0,
  aagun = FALSE,
```

```
  bombarding_battleships = 0,
  bombarding_cruisers = 0,
  write_to_console = TRUE
)
```

Arguments

offense_infantry	Number of infantry on the attacking side, which hit when the dice roll is a 1
offense_artillery	Number of artillery on the attacking side, which hit when the dice roll is a 2 or less
offense_tanks	Number of tanks on the attacking side, which hit when the dice roll is a 3 or less
offense_fighters	Number of fighters on the attacking side, which hit when the dice roll is a 3 or less
offense_bombers	Number of bombers on the attacking side, which hit when the dice roll is a 4 or less
defense_infantry	Number of infantry on the defending side, which hit when the dice roll is a 2 or less
defense_artillery	Number of artillery on the defending side, which hit when the dice roll is a 2 or less
defense_tanks	Number of tanks on the defending side, which hit when the dice roll is a 3 or less
defense_fighters	Number of infantry on the defending side, which hit when the dice roll is a 4 or less
defense_bombers	Number of infantry on the defending side, which hit when the dice roll is a 1 or less
aagun	Whether or not an Anti-Air gun is present, which rolls one time at the beginning of the battle for each attacking aircraft, and hits if the roll is a 1
bombarding_battleships	Number of bombarding battleships, which bombard at the start of the battle, and hit at 4 or less
bombarding_cruisers	Number of bombarding cruisers, which bombard at the start of the battle, and hit at 3 or less
write_to_console	If true, writes the output to the console, if false, returns as a vector

Value

Offense Loses or Defense Loses and remaining units in lines of text

Examples

```
land_battle(offense_tanks = 4, offense_fighters = 3, defense_tanks = 9, aagun = TRUE)
```

`land_simulate` *Simulate Land Battles*

Description

Simulates a number of land battles and gives the percentage of them won and lost. Use to find the probability of winning a particular land battle.

Usage

```
land_simulate(
  offense_infantry = 0,
  offense_artillery = 0,
  offense_tanks = 0,
  offense_fighters = 0,
  offense_bombers = 0,
  defense_infantry = 0,
  defense_artillery = 0,
  defense_tanks = 0,
  defense_fighters = 0,
  defense_bombers = 0,
  aagun = FALSE,
  bombarding_battleships = 0,
  bombarding_cruisers = 0,
  sample_size = 10000,
  decimals = 1,
  write_to_console = TRUE
)
```

Arguments

`offense_infantry`
: Number of infantry on the attacking side, which hit when the dice roll is a 1

`offense_artillery`
: Number of artillery on the attacking side, which hit when the dice roll is a 2 or less

`offense_tanks`
: Number of tanks on the attacking side, which hit when the dice roll is a 3 or less

`offense_fighters`
: Number of fighters on the attacking side, which hit when the dice roll is a 3 or less

`offense_bombers`
: Number of bombers on the attacking side, which hit when the dice roll is a 4 or less

`defense_infantry`
: Number of infantry on the defending side, which hit when the dice roll is a 2 or less

`defense_artillery`
: Number of artillery on the defending side, which hit when the dice roll is a 2 or less

`defense_tanks`
: Number of tanks on the defending side, which hit when the dice roll is a 3 or less

`defense_fighters`
: Number of infantry on the defending side, which hit when the dice roll is a 4 or less

`defense_bombers`
: Number of infantry on the defending side, which hit when the dice roll is a 1 or less

`aagun`
: Whether or not an Anti-Air gun is present, which rolls one time at the beginning of the battle for each attacking aircraft, and hits if the roll is a 1

`bombarding_battleships`
: Number of bombarding battleships, which bombard at the start of the battle, and hit at 4 or less

`bombarding_cruisers`
: Number of bombarding cruisers, which bombard at the start of the battle, and hit at 3 or less

`sample_size`
: Number of land battles simulated

`decimals`
: Number of decimal places the percentages are rounded to

`write_to_console`
: If true, writes the output to the console, if false, returns as a vector

Value

Percentage of the land battles won and lost.

Examples

```
land_simulate(offense_infantry = 10, defense_infantry = 6, decimals = 2)
```

---

raid_battle *Strategic Bombing Raid*

Description

Simulates one strategic bombing raid on an enemy industrial complex

Usage

```
raid_battle(
  offense_fighters = 0,
  offense_bombers = 1,
  defense_fighters = 0,
  aagun = FALSE,
  write_to_console = TRUE
)
```

Arguments

offense_fighters	Number of fighters brought to the strategic bombing raid if using optional rules for strategic bombing raids which include fighters
offense_bombers	Number of bombers brought to bombing raid
defense_fighters	Number of fighters defending in the strategic bombing raid if using optional rules for strategic bombing raids which include fighters
aagun	Is an anti aircraft gun present on the defending side
write_to_console	If true, writes the output to the console, if false, returns as a vector

Value

IPC Damage done by strategic bombing raid to industrial complex, number of offense fighters left if using optional rules, number of bombers left, number of defense fighters left if using optional rules in lines of text

Examples

```
raid_battle(offense_bombers = 3, aagun = TRUE)
```

---

sea_round *Sea Round*

Description

Simulates one round of sea combat

Usage

```
sea_round(
  offense_submarines = 0,
  offense_destroyers = 0,
  offense_carriers = 0,
```

```
  offense_cruisers = 0,
  offense_battleships = 0,
  offense_fighters = 0,
  offense_bombers = 0,
  defense_submarines = 0,
  defense_destroyers = 0,
  defense_carriers = 0,
  defense_cruisers = 0,
  defense_battleships = 0,
  defense_fighters = 0,
  write_to_console = TRUE
)
```

Arguments

`offense_submarines`
: Number of submarines on the attacking side, which hit when the dice roll is a 2 or less

`offense_destroyers`
: Number of destroyers on the attacking side, which hit when the dice roll is a 2 or less

`offense_carriers`
: Number of carriers on the attacking side, which hit when the dice roll is a 1

`offense_cruisers`
: Number of cruisers on the attacking side, which hit when the dice roll is a 3 or less

`offense_battleships`
: Number of battleships on the attacking side, which hit when the dice roll is a 4 or less

`offense_fighters`
: Number of fighters on the attacking side, which hit when the dice roll is a 3 or less

`offense_bombers`
: Number of bombers on the attacking side, which hit when the dice roll is a 4 or less

`defense_submarines`
: Number of submarines on the defending side, which hit when the dice roll is a 1

`defense_destroyers`
: Number of destroyers on the defending side, which hit when the dice roll is a 2 or less

`defense_carriers`
: Number of carriers on the defending side, which hit when the dice roll is a 2 or less

`defense_cruisers`
: Number of cruisers on the defending side, which hit when the dice roll is a 3 or less

`defense_battleships`
: Number of battleships on the defending side, which hit when the dice roll is a 4 or less

`defense_fighters`
: Number of fighters on the defending side, which hit when the dice roll is a 4 or less

`write_to_console`
: If true, writes the output to the console, if false, returns as a vector

Value

Number of offensive air hits, offensive submarine hits, offensive other hits, defensive air hits, defensive submarine hits, and defensive other hits in several lines of text

Examples

```
sea_round(offense_submarines = 1, offense_bombers = 1, defense_battleships = 1)
```

Index