Poker hand simulation - single pair probability

Global parameters

In [12]:
```
sample_size = 30
simulation_size = 1000
poker_hand_size = 5
confidence = 0.95
```

Create class for deck cards

In [13]:
```
from enum import Enum

class Suit(Enum):
    __order__ = 'CLUBS DIAMONDS HEARTS SPADES'
    CLUBS = 1
    DIAMONDS = 2
    HEARTS = 3
    SPADES = 4

class Card:
    def __init__(self, value, suit):
        self.value = value
        self.suit = suit
```

Create deck

In [14]:
```
deck = []
for v in range(13):
    for s in Suit:
        deck.append(Card(v, s))
```

Declare simulation function

In [15]:
```
import random

def runSimulation():

    single_pairs = 0

    for n in range(simulation_size):

        #shuffle deck
        random.shuffle(deck)

        #count pairs in hand (top x cards of deck)
        pairs = 0
        for i in range(poker_hand_size):
            for j in range(i + 1, poker_hand_size):

                card1 = deck[i].value
                card2 = deck[j].value

                if card1 == card2:
                    pairs+=1

        #count if single pair
        if pairs == 1:
            single_pairs+=1

    probability = single_pairs / simulation_size

    return probability
```

Run simulations

In [16]:
```
data = []
for i in range(sample_size):
    probability = runSimulation()
    data.append(probability)
```

Print statistics

In [17]:
```
import pandas as pd

df = pd.DataFrame(data)
df.describe()
```

Out[17]:

	0
count	30.000000
mean	0.429467
std	0.015231
min	0.403000
25%	0.416500
50%	0.430500
75%	0.438000
max	0.467000

In [18]:
```
import numpy as np
import scipy.stats

a = 1.0 * np.array(data)
m, se = np.mean(a), scipy.stats.sem(a)
h = se * scipy.stats.t.ppf((1 + confidence) / 2., sample_size - 1)
print("confidence interval is: [" + str(m - h) + ", " + str(m + h)
+ "] at " + str(int(100 * confidence)) + "%")
```

```
confidence interval is: [0.4237793392715345, 0.43515399406179894]
at 95%
```

Print global parameters

In [19]:
```
print("confidence: " + str(confidence))
print("sample_size: " + str(sample_size))
print("simulation_size: " + str(simulation_size))
print("poker_hand_size: " + str(poker_hand_size))
```

```
confidence: 0.95
sample_size: 30
simulation_size: 1000
poker_hand_size: 5
```