

Forum:[Hack Forums](#)

Thread: [Humans Suck at Entropy \(Man vs Machine\)](#)

ImportUsername - October 25, 2013, 2:15 pm

People are bad at entropy (randomness). This code asks the user to try to input 1's and 0's as randomly as possible. The computer will make an educated guess on what you will enter next, based on numbers you've already input. You are predictable.

Anyway, the math behind this is not my own, I translated the code from BASIC from <http://www.loper-os.org/bad-at-entropy/manmach.html>

Here is the code:

<http://pastebin.com/Rfk4YuNm> for 2.x

OR

Spoiler ([Click to View](#))

Code ##

```
#  -- MAN vs MACHINE -- #
# #
# People are not particularly good at entropy (randomness) #
# #
# In this game you try to randomly say 1's and 0's #
# The computer can pass (not guess), or take a guess at what you will say #
# If the computer gets it correct, they get a point, if not, you get a point #
# The computer makes an educated guess based on past inputs #
# #
# This code was translated from the original source code available at: #
# http://www.loper-os.org/bad-at-entropy/manmach.html #
# #
# This code was translated by ImportUsername #
##
```

```
import math
```

```
def new_m():
    m = []
    for v in range(2):
        m.append([])
    for w in range(3):
        m[v].append([])
    for x in range(2):
        m[v][w].append([])
    for y in range(3):
        m[v][w][x].append(0)
    return m
```

```
def main():
    m = new_m()
```

```

i = 1
sp = 0
sk = 0
k1 = 0
k2 = 0
p1 = 0
p2 = 0
n1 = 0
r = 0
while True:
print("--\n\nRound: "+str(i) + \
"\n- Computer Score: "+ str(sk) + \
"\n- Player Score: " + str(sp))
if r != 0: print("\n- Computer Accuracy: " + str((float(sk)/float(r))*100) + "%")
if i < 3:
    n1 = 2
if abs(m[p1][k1][p2][k2][0] - m[p1][k1][p2][k2][1]) <= 1.8 * math.pow(1.01,i):
    n1 = 2
elif m[p1][k1][p2][k2][0] < m[p1][k1][p2][k2][1]:
    n1 = 1
else:
    n1 = 0
n = 2
while n != 1 and n != 0:
    try:
        n = int(raw_input("\n--\nEnter a number 1 or 0: "))
    except:
        print("\nPlease enter either 1 or 0")
if n1 == 1 or n1 == 0:
    print("--\nComputer guesses "+str(n1))
else:
    print("--\nComputer passes")
if i > 2:
    m[p1][k1][p2][k2][n] = m[p1][k1][p2][k2][n] + math.pow(1.1,i)
k1 = k2
k2 = n1
p1 = p2
p2 = n
i = i + 1
if n1 != 2:
    r += 1
    if n == n1:
        sk = sk + 1
    else:
        sp = sp + 1

if __name__ == '__main__':
    main()

```

Later going to make a PyGame version of this similar to the site's version, so the user doesn't need to hit enter between numbers.?

EDIT: clarity

glob - October 25, 2013, 5:29 pm

That code looks extremely messy ._.

I like the concept though. Although it's simple, and there is a 50% chance of success.

ImportUsername - October 25, 2013, 5:36 pm

I didn't fully understand the code when writing it, just trying to translate it to work, so it won't look the best it could :/ (Just imagine trying to translate a story from russian to english for example)

I don't get what you mean about a 50% chance of success. It's not guessing randomly on what you will type next. It's actually using the numbers you've already entered to make an educated guess at what you will type next. Pretty much always gets above 50%, if not 60 or 70%. If I had wanted something to just guess randomly I could have done it in under 10 lines of code. This is different

glob - October 25, 2013, 5:54 pm

(Oct 25, 2013 05:36 PM)ImportUsername Wrote: [\[CITE\]](#) I didn't fully understand the code when writing it, just trying to translate it to work, so it won't look the best it could :/ (Just imagine trying to translate a story from russian to english for example)

I don't get what you mean about a 50% chance of success. It's not guessing randomly on what you will type next. It's actually using the numbers you've already entered to make an educated guess at what you will type next. Pretty much always gets above 50%, if not 60 or 70%. If I had wanted something to just guess randomly I could have done it in under 10 lines of code. This is different

I didn't really explain that well. Yes there is math involved and the percentage would be higher, but I guess the point I was trying to make is that once you get past all that there are only two options that it could guess, therefore making it a somewhat educated 50%

EDIT: Yeah I have no idea where I was going with that. Ignore me.

ImportUsername - October 25, 2013, 5:56 pm

The point being made with the program though is that YOU are not random. It would be 50% if you were but you aren't. People are bad at that. Try it out, if not with this program then go to the website for 1 minute. Type in what seems like a random series of zeroes and ones, and you may be surprised just how often it guesses it - <http://www.loper-os.org/bad-at-entropy/manmach.html>

glob - October 25, 2013, 6:29 pm

(Oct 25, 2013 05:56 PM)ImportUsername Wrote: [\[CITE\]](#) The point being made with the program though is that YOU are not random. It would be 50% if you were but you aren't. People are bad at that. Try it out, if not with this program then go to the website for 1 minute. Type in what seems like a random series of zeroes and ones, and you may be surprised just how often it guesses it -

<http://www.loper-os.org/bad-at-entropy/manmach.html>

With two options it's very hard to be random without subliminally falling into a pattern yes, but that is in part due to the extreme limitation of only having two options. Just saying that it's relatively simple to guess after a bit of pattern analysis.

Wudizit90 - October 25, 2013, 8:44 pm

I agree this is not a good measure of a humans ability to be random.

BeerMonger - October 26, 2013, 4:22 am

I'm aware of the fact that it is based on math, but still looks pretty 50/50 to me. (60-34 score so far ^^)

ImportUsername - October 26, 2013, 6:52 am

(Oct 26, 2013 04:22 AM)BeerMonger Wrote: [\[CITE\]](#) I'm aware of the fact that it is based on math, but

still looks pretty 50/50 to me. (60-34 score so far ^^)

It doesn't get 100%, it just can't be THAT accurate, but it normally sits around 66%. It will do better if you do it as quick as possible without thinking about it, because then you really set into a pattern. And 60-34 isn't that close to 50-50?

BeerMonger - *October 27, 2013, 1:16 am*

Well.. since you have only 2 options to choose from its a bit hard to be unexpected..

You only have 4 options, a sequence of following 0s/1s or 1 followed by 0 and vice versa..
but yeah its nice any how P:

ImportUsername - *October 27, 2013, 5:11 am*

(Oct 27, 2013 01:16 AM)BeerMonger Wrote: [\[CITE\]](#) Well.. since you have only 2 options to choose from its a bit hard to be unexpected..

You only have 4 options, a sequence of following 0s/1s or 1 followed by 0 and vice versa..
but yeah its nice any how P:

The point of the the little game is to show that we're bad at being 'unexpected'. If we could be perfectly random, then even with only 2 options the computer in this game would only guess about 50%

Misingnoglic - *October 31, 2013, 3:12 pm*

Can someone explain to me what is actually happening here code wise?