# Assignment 1

CSCI2244-Randomness and Computation
Due Wednesday, January 24, at 11:30PM

This assignment is an empirical exploration of runs, or 'streaks', in coin-tossing data. This is precisely the feature that made it possible to distinguish genuine coin-tossing data from made-up data in our first-day exercise. The problems are intended both to get you thinking about probability-related questions before we dive into the formal mathematical treatment, and to give you a good workout with the plotting features of matplotlib.

There is more coding than usual in this assignment, but less than meets the eye. You have to write a number of different Python functions, but they are all quite short and very similar to one another. You are then asked to answer some questions based on execution of your programs. The last problem asks you to analyze some sports data.

Please read carefully the instructions on assignment submissions—your assignment should include not just the code, but nice-looking graphs, and thorough, careful answers to all the questions.

## 1 Problem 1: Run lengths

Write a Python function

```
runlengths(p,n)
```

where `p` is a float between 0 and 1, and `n` is a nonnegative int, that simulates $n$ tosses of a coin whose probability of heads is $p$, and returns a list of the run lengths. For example, if the simulated sequence of coin tosses is

$$HTTHHTHHHTTHHTTTTH$$

then the list of run lengths returned is [1,2,2,1,3,2,2,4,1].

HINT: Your function should not explicitly return or print the sequence of coin tosses itself. But you will have to generate this sequence internally, and use it to calculate the sequence of run lengths. You *should* print the sequence of coin tosses while you are testing and debugging (for small values of n) to make sure your function does the right thing. (For instance, it is very easy to forget to include the last run.)

We will be interested in the maximum run length (4 in the example above) and the number of runs (9, in the example above). These are given by

```
max(runlengths(p,n))
```

and

```
len(runlengths(p,n))
```

You may find it helpful to turn these into functions `maxrunlength(p,n)` and `numruns(p,n)`, although this is not necessary.

# 2 Problem 2: Histogram of maximum run lengths

Write a function

```
histmaxruns(p,n,numtrials,cum=False)
```

that performs `numtrials` simulations of $n$ tosses of a coin whose probability of heads is $p$, collects the maximum run length for each trial, and plots a histogram of the result. So, for instance, if the most frequently occurring maximum run length in 1000 trials of 50 tosses of a fair coin is 5, then the tallest column in the histogram will be at $x = 5$. You may have to play around with `xlim`, or the bins argument to `hist`, adjusting the limits of the $x$-coordinate variable in the plot, to get the best display, since this will vary for different choices of $p$ and $n$. If the optional argument `cum` is set to `True`, then the function should display a

cumulative histogram. The text in the plot should display the values of `p, n, numtrials` in a clear fashion.

(a) Use this function to determine the most likely length of the longest run in 200 tosses of a fair coin. (In this and other parts of the problem, provide the plot to support your answer). To orient you, the fakers in the in-class experiment all had maximum run length 5 on 200 tosses, while the genuine coin-tossers got maximum run lengths of 6,7 and 8.

(b) Same question, but now suppose that it is an unfair coin, with heads probability 0.6.

(c) Using the result of the simulation, estimate the probability that in 200 tosses of a fair coin, there is no run of length greater than 5. (It is easier to do this if you display the cumulative histogram.)

# 3 Problem 3: Histogram of number of run lengths

Write a function

```
histnumruns(p,n,numtrials,cum=False)
```

that performs `numtrials` simulations of $n$ tosses of a coin whose probability of heads is $p$, collects the number of runs in each trial, and plots a histogram of the result.

(a) Use this to determine the most likely number of runs in 50 tosses of a fair coin.

(b) Now run this function with p=0.2, n=50, and numtrials=1000. You should see some rather striking behavior in the plot. What is it? Why does it occur? (The behavior itself will jump right out at you, but the explanation is a bit tricky.)

# 4 Average number of runs, and the 'hot hand'

(a) Write a function `plotnumruns()` that produces a single plot: For $n = 0, 10, 20, ..., 400$, perform $n$ tosses of a coin (with heads probability $p$) 1000 times, and plot the average number of runs for each value of $n$. For example, if $n = 100$, then the program will perform 100 tosses of the coin 1000 times, and the height of the plot at $x = n$ will be the average of the number of runs over all 1000 trials.

Your plot should superimpose three line plots, for $p = 0.3, 0.4, 0.5$.

(b) Just as a reality check, the plots in part (a) should be close to straight lines. It is not hard to see why—if you do twice as many coin tosses, you should get approximately twice as many runs. In other words, for fixed $p$ and moderately large values of $n$, the average number of runs on $n$ tosses divided by $n$ should be constant. Using 1000 trials and 200 tosses for each $p$, plot this constant for $p = 0, 0.05, 0.1, \ldots, 1$.

Soon we will learn a simple way to compute this constant. If you did this correctly, you should find that your plot is symmetrical about $x = 0.5$—why is this?

(c) Sports fans, sports commentators, and players themselves, generally believe that players' performances exhibit remarkable streaks. A famous 1985 paper by psychologists John Gilovich, Robert Vallone, and Amos Tversky, asked whether this perception was an illusion. The problem might be posed this way: Suppose a basketball player succeeds in 60% of his shots. If the player's performance is unusually 'streaky', then the run lengths should in general be longer than those that we expect from tosses of a coin that comes up heads 60% of the time, and the number of runs should consequently be smaller than what we see with the coin. On the other hand, if the player's streak behavior is no different from that of the coin, then we would have no more reason for saying that the player is 'on fire' than for saying 'this nickel is on fire'. As Gilovich, et. al., put it:

> Consider a professional basketball player who makes 50% of his shots. This player will occasionally hit four or more shots in a row. Such runs can properly be called streak shooting, however, only if their length or frequency exceeds what is expected on the basis of chance alone. The player's performance, then, can be compared to a sequence of hits and misses generated by tossing a coin. A player who produces longer sequences of hits than those produced by tossing a coin can be said to have a 'hot hand', or be described as a 'streak shooter'.

Based on their analysis, the psychologists concluded that in general, players' streak behavior really was not different from that of the coin, and the the hot hand is largely an illusion. This has been much discussed in the more than thirty years since the paper was published, and both their methods and conclusions have been widely criticized.

Be that as it may, in this problem you are asked to use the tools you developed above for studying run behavior in coins to analyze some of the data from the

| Player | Hits | Misses | Number of Runs |
|---|---|---|---|
| 1 | 54 | 46 | 56 |
| 2 | 35 | 65 | 46 |
| 3 | 60 | 40 | 40 |
| 4 | 36 | 54 | 47 |
| 5 | 42 | 58 | 55 |
| 6 | 57 | 43 | 41 |
| 7 | 42 | 33 | 31 |
| 8 | 25 | 25 | 27 |
| 9 | 54 | 46 | 32 |
| 10 | 60 | 40 | 51 |
| 11 | 58 | 42 | 48 |
| 12 | 44 | 56 | 52 |
| 13 | 61 | 39 | 52 |
| 14 | 59 | 41 | 50 |

paper. The authors present (along with much other data) the following table, obtained from 14 players on the Cornell basketball team, who made repeated shots from points a fixed distance from the basket. (In most cases, exactly 100 shots.) The table gives the number of hits, misses and the number of runs. As with the coins, a sequence of consecutive hits, as well as a sequence of consecutive misses, counts as a run.

They conclude that the performance of only one of these 14 players deviates significantly from that of a coin with the same percentage of successes. We're not in a position yet to say what constitutes 'significant' deviation, but you should be able to use the results of your simulation to identify the outlier. Just don't tell me who it is; back up your conclusion carefully.