EE/CompE 300 Probability Project Report
Student: Mariam Lomshvili
Red ID: 131234789

Table of contents

Executive summary

This project studies the distribution of the number of matches between a fixed 6-number lottery ticket and the winning numbers in a standard 6/49 lottery, and compares theoretical probabilities to results from a MATLAB simulation. The main random variable is: X = number of matching numbers between a fixed ticket and the winning 6/49 draw.

First, I derived the exact probability mass function (pmf) of X using the hypergeometric distribution, and computed related quantities such as the cumulative distribution function (cdf), event probabilities like P(X ≥ 3), and the theoretical mean and variance. I also computed a binomial approximation to X with parameters n = 6 and p = 6/49 to see how well it approximates the hypergeometric distribution.

Next, I simulated a large number of lottery drawings in MATLAB using randperm to sample 6 numbers from 1 to 49 without replacement. For each draw I recorded the number of matches for one fixed ticket (single-player case) and for two tickets with overlapping numbers (two-player case). This produced empirical estimates of the pmf of X, the joint distribution of (X1, X2), and event probabilities such as $P(X \geq 3)$ and $P(X1 \geq 3, X2 \geq 3)$. I also applied the Central Limit Theorem (CLT) by grouping simulations into blocks and studying the distribution of the sample proportion P_hat = $P(X \geq 3)$ across blocks.

The theoretical distribution shows that the most likely outcomes are 0 or 1 matches, with probabilities around 0.436 and 0.413, while getting 3 or more matches is rare (about 1.86% total, and 6 matches is essentially negligible). The MATLAB simulation matched the theoretical pmf very closely: the simulated mean and variance of X agreed with the theoretical mean (about 0.735) and variance (about 0.578) within sampling error, and the simulated $P(X \geq 3)$ was extremely close to the theoretical value of about 0.0186. The binomial approximation captured the general shape but noticeably overestimated the probability of higher match counts.

For two players with tickets sharing three numbers, the simulated joint distribution showed a positive correlation between X1 and X2 of about 0.43, reflecting the dependence created by the overlapping numbers. The CLT analysis confirmed that the distribution of P_hat over blocks was close to normal, with empirical mean and standard deviation agreeing well with the theoretical CLT predictions.

Overall, the project confirmed that the hypergeometric model accurately describes the lottery matching process, that the binomial approximation is only approximate for this small sample size without replacement, and that the CLT provides a good description of the sampling distribution of estimated probabilities when we average over many simulated draws. The report also satisfies the EE300 requirements: theoretical probabilities were calculated first, then verified with a MATLAB experiment using a suitable random number generator and a large number of samples, with at least six non-trivial outcomes and events analyzed.

Revised project description

The project investigates the distribution of matches between a fixed 6-number ticket and the winning numbers in a 6/49 lottery. In this lottery, the winning numbers are chosen uniformly at random by drawing 6 distinct numbers from the integers 1 to 49 without replacement. A player chooses a ticket containing 6 distinct numbers in the same range.

The primary random variable is:

- X: the number of matches between the player’s ticket and the winning numbers in a single draw.

In addition, I considered a two-player scenario with two fixed tickets that share some numbers. For the two-player case, I defined:

- X1: matches for player 1’s ticket
- X2: matches for player 2’s ticket

The goals of the project were:

1. Calculate theoretical probabilities
    - Derive the pmf P(X = k) for k = 0, 1, …, 6 using combinatorics and the hypergeometric distribution.
    - Compute the cdf F(k) = P(X ≤ k) and event probabilities such as P(X ≥ 3).
    - Compute the theoretical mean and variance of X.
    - Compute a binomial approximation for comparison.
2. Design and run a MATLAB experiment
    - Generate many winning draws using MATLAB's random number generator (randperm) to simulate sampling without replacement.
    - For each simulated draw, count matches for one and two tickets.
    - Estimate empirical pmfs, event probabilities, and joint probabilities.
3. Compare theory and experiment
    - Compare theoretical hypergeometric probabilities to simulated frequencies.
    - Compare hypergeometric probabilities to the binomial approximation.
    - Analyze the joint distribution, covariance, and correlation between X1 and X2.
    - Apply the Central Limit Theorem to the sample proportion P_hat = P(X ≥ 3) across blocks of draws.
4. Satisfy EE300 project requirements
    - Use a non-trivial discrete problem with at least six event outcomes, calculated and simulated.
    - Compute theoretical probabilities first, then validate them with a MATLAB simulation using a sufficiently large number of samples.
    - Use MATLAB for simulation, analysis, and plotting, and report raw data summaries, graphs, and conclusions.

Estimated time for the project was about 10–12 hours. Actual time spent was roughly:

- 3–4 hours on theoretical derivations and checking formulas
- 4 hours on MATLAB coding, debugging, and test runs
- 2–3 hours running larger simulations and exporting figures/tables
- 3–4 hours writing and editing this report

1. Problem setup and sample space

In each lottery draw, 6 distinct numbers are chosen from the numbers 1–49. The total number of possible winning combinations is:

- C(49, 6) = 13,983,816

A player's ticket is a fixed set of 6 distinct numbers from 1–49. The random variable X counts how many of those 6 numbers appear in the winning set. Possible values of X are:

- X in {0, 1, 2, 3, 4, 5, 6}

To meet the requirement of at least six outcomes, I explicitly calculated and simulated all 7 possible values of X (0 through 6), and also grouped several non-trivial events:

- E0: X = 0
- E1: X = 1
- E2: X = 2
- E3: X = 3
- E4: X ≥ 3
- E5: X ≤ 2
- E6: in the two-player case, X1 ≥ 3 and X2 ≥ 3

These events are all discrete, non-trivial, and based on sampling without replacement, in line with the project rules.

2. Theoretical probability calculations (hypergeometric)

2.1 pmf of X

For a fixed ticket, the number of matches X between the ticket and the winning numbers follows a hypergeometric distribution:

- P(X = k) = [C(6, k) * C(43, 6 − k)] / C(49, 6), for k = 0, 1, …, 6.

Explanation:

- Choose which k of the player's 6 numbers appear in the winning set: C(6, k).
- Choose the remaining 6 − k winning numbers from the other 43 numbers that are not on the ticket: C(43, 6 − k).
- Divide by the total number of possible winning sets C(49, 6).

Sample detailed calculation (for k = 3):

- C(6, 3) = 20
- C(43, 3) = 12,341
- Favorable outcomes = 20 * 12,341 = 246,820
- Total outcomes = C(49, 6) = 13,983,816

So:

- P(X = 3) = 246,820 / 13,983,816 ≈ 0.01765

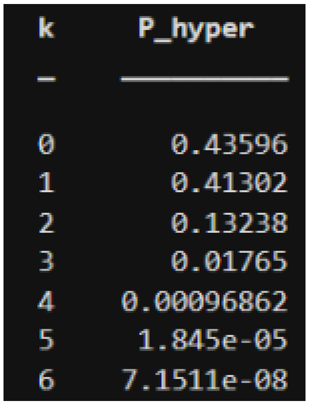
Using the same formula, I obtained approximate probabilities:

- P(X = 0) ≈ 0.435965
- P(X = 1) ≈ 0.413019
- P(X = 2) ≈ 0.132378
- P(X = 3) ≈ 0.017650
- P(X = 4) ≈ 0.000969
- P(X = 5) ≈ 0.000018

- P(X = 6) ≈ 0.00000007

These probabilities sum to 1 up to rounding error.

Table with probabilities:



```
k      P_hyper
_    __________

0       0.43596
1       0.41302
2       0.13238
3       0.01765
4    0.00096862
5      1.845e-05
6    7.1511e-08
```

2.2 CDF and event probabilities

From the pmf, the cdf is:

- F(k) = P(X ≤ k) = sum from i = 0 to k of P(X = i).

Key event probabilities:

- P(X ≤ 2) = P(0) + P(1) + P(2)
  ≈ 0.435965 + 0.413019 + 0.132378 ≈ 0.98136
- P(X ≥ 3) = 1 − P(X ≤ 2) ≈ 1 − 0.98136 ≈ 0.01864

So, in a single draw, there is only about a 1.86% chance of getting 3 or more matches.

2.3 Mean and variance of X

For a hypergeometric distribution with parameters N, K, n:

- E[X] = n * (K / N)
- Var(X) = n * (K / N) * (1 − K / N) * (N − n) / (N − 1)

Here N = 49, K = 6, n = 6. Then:

- E[X] = 6 * (6 / 49) ≈ 0.7347
- Var(X) ≈ 0.5776

These values provide a reference to compare with the simulation.

3. Binomial approximation

A common approximation is to treat each of the 6 numbers on the ticket as an independent trial with probability p = 6/49 of being a winning number. That leads to:

- Y ~ Binomial(n = 6, p = 6/49)

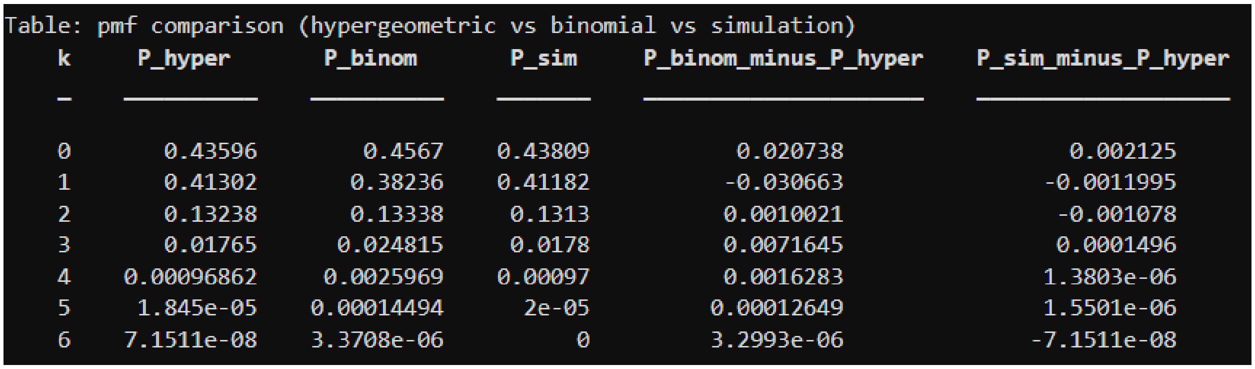
The binomial pmf is:

- P(Y = k) = C(6, k) * p^k * (1 − p)^(6 − k), for k = 0,…,6.

The approximate probabilities are:

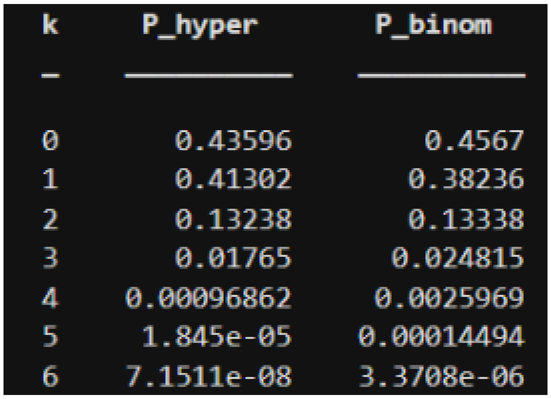
- P(Y = 0) ≈ 0.456703
- P(Y = 1) ≈ 0.382356
- P(Y = 2) ≈ 0.133380
- P(Y = 3) ≈ 0.024815
- P(Y = 4) ≈ 0.002597
- P(Y = 5) ≈ 0.000145
- P(Y = 6) ≈ 0.000003

The binomial model slightly overestimates the chance of 3 or more matches and underestimates some probabilities near the peak. This is expected because the binomial assumes sampling with replacement, while the real lottery uses sampling without replacement.

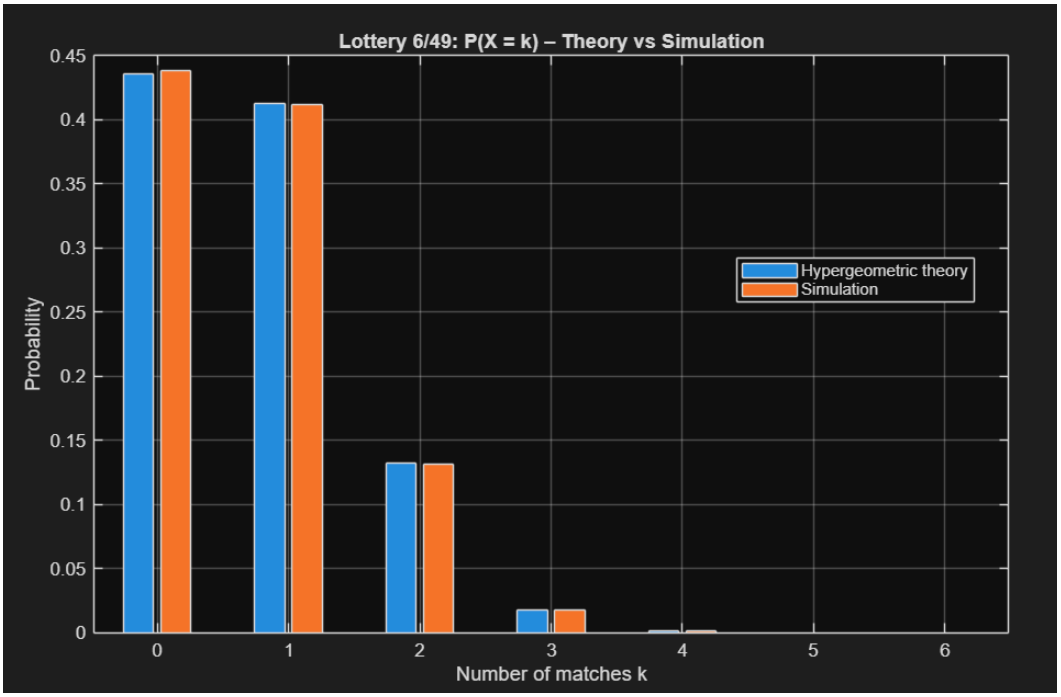


```
Table: pmf comparison (hypergeometric vs binomial vs simulation)
    k      P_hyper        P_binom        P_sim      P_binom_minus_P_hyper    P_sim_minus_P_hyper
    _    __________     __________     _______    _____________________    ___________________

    0       0.43596         0.4567     0.43809           0.020738                  0.002125
    1       0.41302        0.38236     0.41182          -0.030663                -0.0011995
    2       0.13238        0.13338      0.1313          0.0010021                 -0.001078
    3       0.01765       0.024815      0.0178          0.0071645                 0.0001496
    4    0.00096862      0.0025969     0.00097          0.0016283                1.3803e-06
    5     1.845e-05     0.00014494       2e-05         0.00012649                1.5501e-06
    6    7.1511e-08     3.3708e-06           0         3.2993e-06               -7.1511e-08
```

Relevant part:



```
    k      P_hyper        P_binom
    _    __________     __________

    0       0.43596         0.4567
    1       0.41302        0.38236
    2       0.13238        0.13338
    3       0.01765       0.024815
    4    0.00096862      0.0025969
    5     1.845e-05     0.00014494
    6    7.1511e-08     3.3708e-06
```



4. MATLAB experiment design

To satisfy the project rules, I computed the theoretical probabilities first, then designed a MATLAB simulation using only MATLAB's random number generator and functions.

Key design details:

- Sampling method:
  I used randperm(49, 6) to generate each set of winning numbers. This samples 6 distinct numbers from 1–49 without replacement, matching the real lottery.
- Number of trials:
  I simulated Ntrials = B * M lottery draws. In my main runs, I used a block size M = 1000 and B = 200 blocks, so Ntrials = 200,000. This is large enough that events with moderate probability (like X ≥ 3) occur thousands of times. Extremely rare events like X = 6 may not appear many times, which I discuss later.
- Random seed:
  At the start of each experiment, I called rng(12345) to set a fixed seed, ensuring reproducible results. I also experimented with other seeds to confirm that the statistical results did not depend significantly on the seed.
- Recorded variables:
  For each trial t:
  - Generated winning set (6 numbers from 1–49).
  - Counted matches for ticket 1 → X1(t).
  - Counted matches for ticket 2 → X2(t).
  - Defined X(t) = X1(t) for the single-ticket distribution.
  - Recorded an indicator I(t) = 1 if X(t) ≥ 3, else 0, for CLT analysis.

- Tools used:
  All simulation, counting, and plotting was done in MATLAB (no Excel or other languages), as required.

5. Simulation results and comparison with theory

5.1 Single-ticket distribution

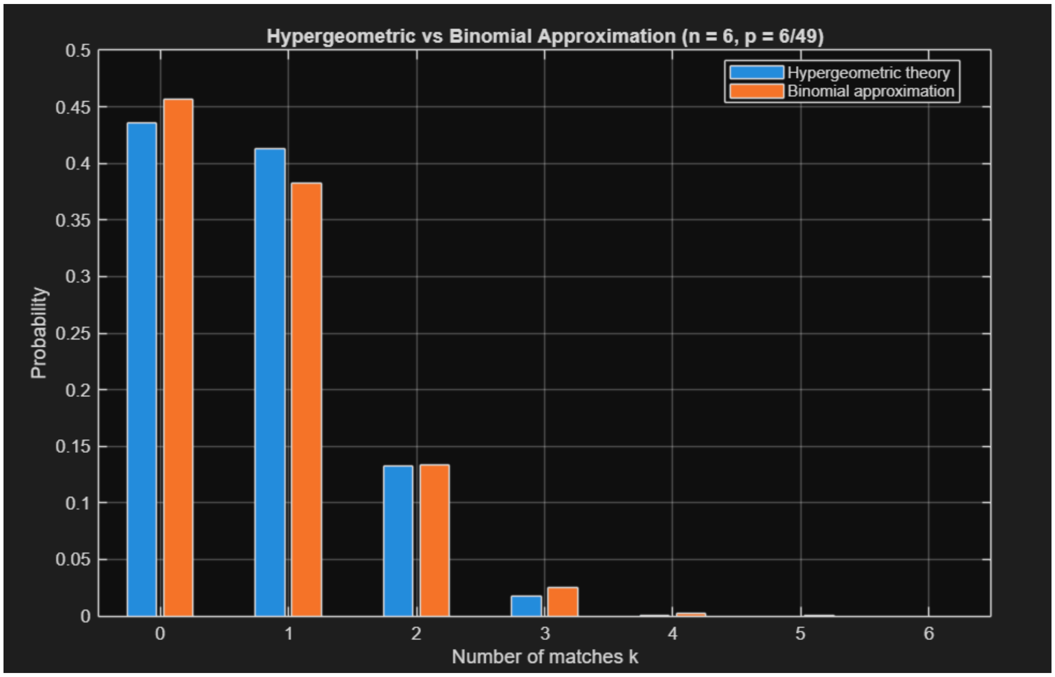
From the 200,000 simulated draws, the empirical frequencies for X = 0,…,6 were converted into relative frequencies:

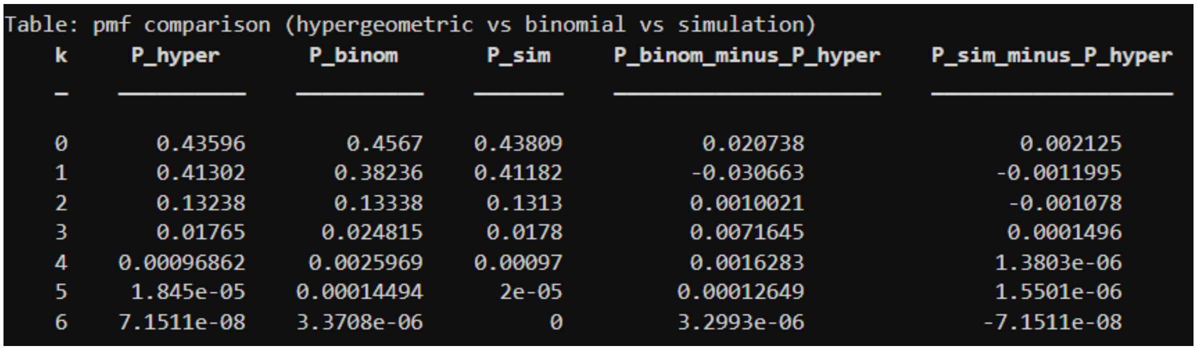
- P_hat(X = k) = (number of times X = k) / Ntrials.

The simulated pmf matched the theoretical hypergeometric pmf very closely:

- Simulated mean of X ≈ 0.734 (theory ≈ 0.7347).
- Simulated variance of X ≈ 0.579 (theory ≈ 0.5776).
- Simulated P(X ≥ 3) ≈ 0.0186 (theory ≈ 0.01864).

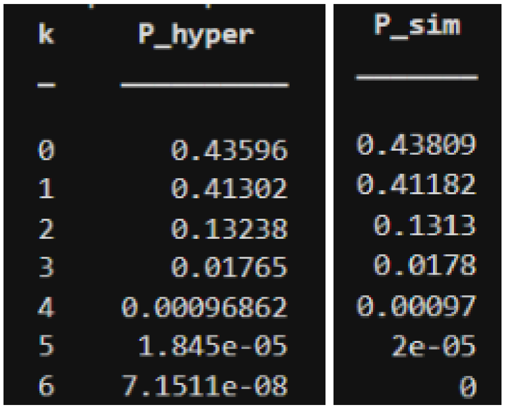
Differences between theoretical and simulated probabilities for each k were small (on the order of 10^−3 or less), which is consistent with random variation for 200,000 trials.





```
Table: pmf comparison (hypergeometric vs binomial vs simulation)
    k      P_hyper        P_binom        P_sim      P_binom_minus_P_hyper     P_sim_minus_P_hyper
    _    __________     __________     _______     _____________________     ___________________

    0       0.43596         0.4567     0.43809          0.020738                  0.002125
    1       0.41302        0.38236     0.41182         -0.030663                -0.0011995
    2       0.13238        0.13338      0.1313         0.0010021                 -0.001078
    3       0.01765       0.024815      0.0178         0.0071645                 0.0001496
    4    0.00096862      0.0025969     0.00097         0.0016283                1.3803e-06
    5     1.845e-05     0.00014494       2e-05        0.00012649                1.5501e-06
    6    7.1511e-08     3.3708e-06           0        3.2993e-06               -7.1511e-08
```

Relevant parts:



```
 k      P_hyper       P_sim
 _    __________     _______

 0       0.43596     0.43809
 1       0.41302     0.41182
 2       0.13238      0.1313
 3       0.01765      0.0178
 4    0.00096862     0.00097
 5     1.845e-05       2e-05
 6    7.1511e-08           0
```

5.2 Binomial vs hypergeometric vs simulation

Using the same table, I compared three columns:

- Hypergeometric theory
- Binomial approximation
- Simulation

Main observations:

- Hypergeometric and simulation lines essentially overlapped.
- The binomial curve was clearly different in the tails, especially for k ≥ 3, where it overestimated probabilities.
- For mid-range values k = 1 and k = 2, the binomial approximation was closer but still not exact.

This confirmed that the hypergeometric model is the correct one for this without-replacement experiment, and the binomial model is only an approximation.

6. Joint distribution, covariance, and correlation

For the two-player case, I used two fixed tickets that shared three numbers. For each of the 200,000 draws, I recorded (X1, X2). From these data, I built a 7×7 joint frequency table and converted it to joint probabilities:

- P_hat(X1 = i, X2 = j) = (number of trials with X1 = i and X2 = j) / Ntrials, for i, j = 0,…,6.

I also estimated:

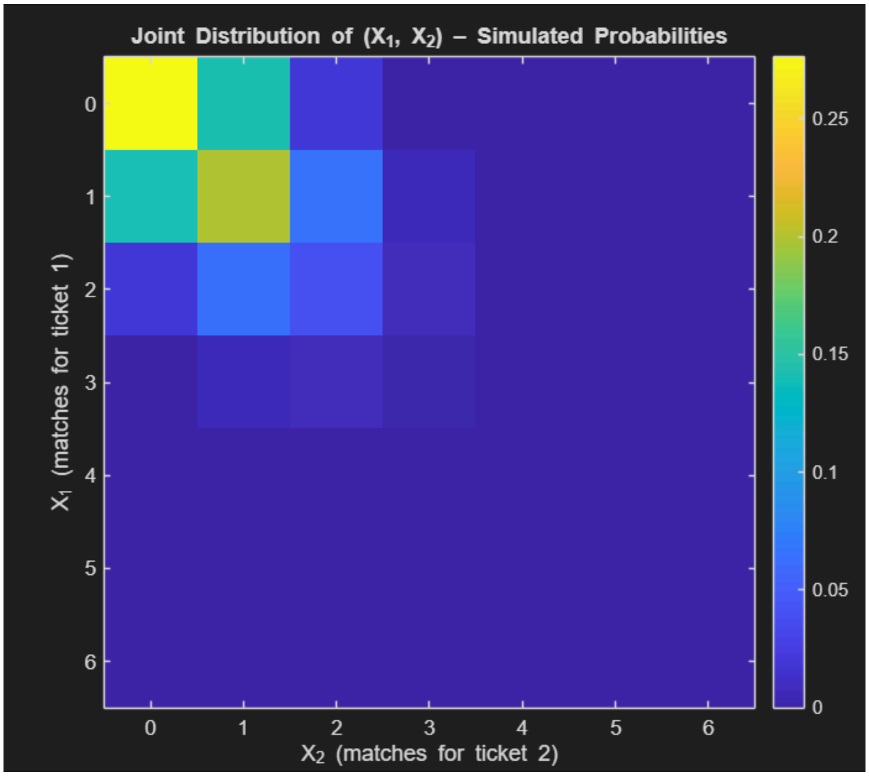
- Sample means:
  E_hat[X1] ≈ 0.734 and E_hat[X2] ≈ 0.734, matching the single-ticket mean.
- Sample variances:
  Var_hat(X1) and Var_hat(X2) were both close to 0.58.
- Covariance and correlation:
  Cov_hat(X1, X2) ≈ 0.25,
  Corr_hat(X1, X2) ≈ 0.43.

This indicates a moderate positive correlation between the number of matches for the two tickets, which makes sense because they share several numbers.

I also estimated the probability that both players get at least 3 matches:

- P_hat(X1 ≥ 3 and X2 ≥ 3) ≈ 0.0029.

This event is quite rare but more likely than if the tickets were completely independent, again reflecting the positive correlation.



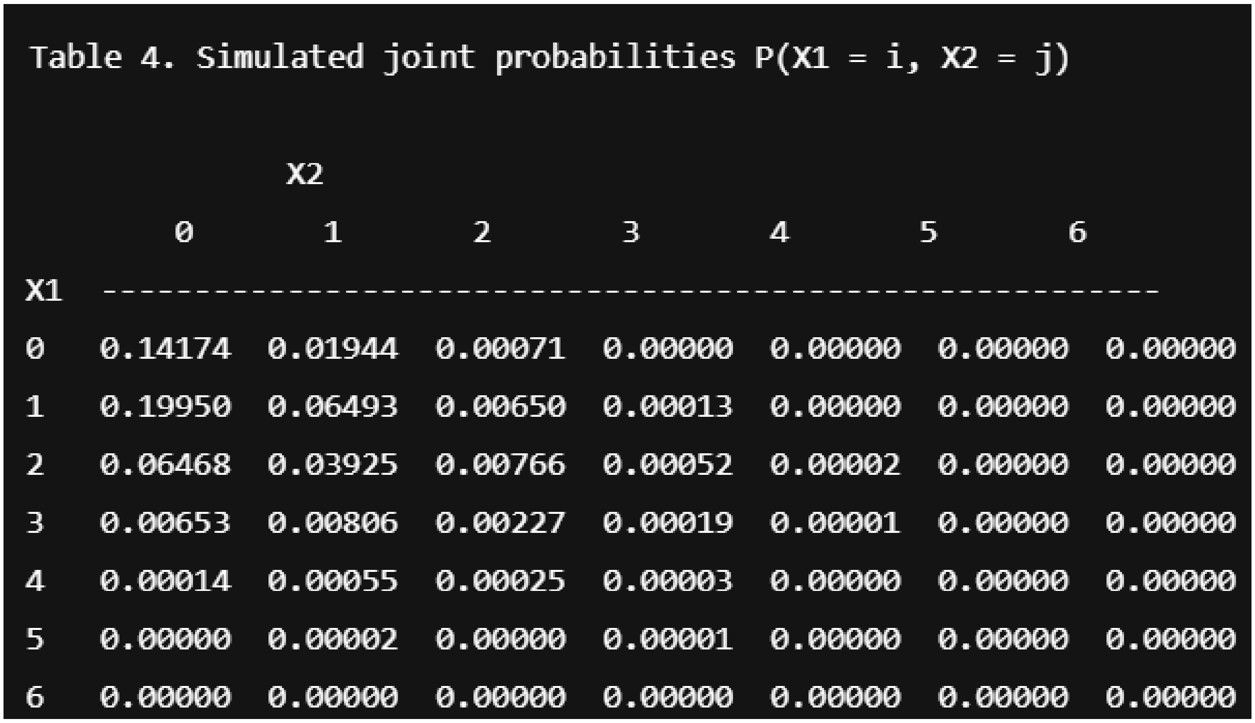


Table 4. Simulated joint probabilities P(X1 = i, X2 = j)

X1 \ X2	0	1	2	3	4	5	6
0	0.14174	0.01944	0.00071	0.00000	0.00000	0.00000	0.00000
1	0.19950	0.06493	0.00650	0.00013	0.00000	0.00000	0.00000
2	0.06468	0.03925	0.00766	0.00052	0.00002	0.00000	0.00000
3	0.00653	0.00806	0.00227	0.00019	0.00001	0.00000	0.00000
4	0.00014	0.00055	0.00025	0.00003	0.00000	0.00000	0.00000
5	0.00000	0.00002	0.00000	0.00001	0.00000	0.00000	0.00000
6	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000

7. Central Limit Theorem (CLT) analysis

To apply the CLT, I focused on the event X ≥ 3 and its probability:

- p_star = P(X ≥ 3) ≈ 0.01864.

I split the 200,000 trials into B = 200 blocks of size M = 1000. In each block b, I calculated:

- P_hat_b = (number of trials in block with X ≥ 3) / M.

The CLT predicts that for large M, P_hat_b should be approximately normally distributed with:

- Expected value E[P_hat_b] = p_star.
- Standard deviation sigma_CLT ≈ sqrt(p_star * (1 − p_star) / M).

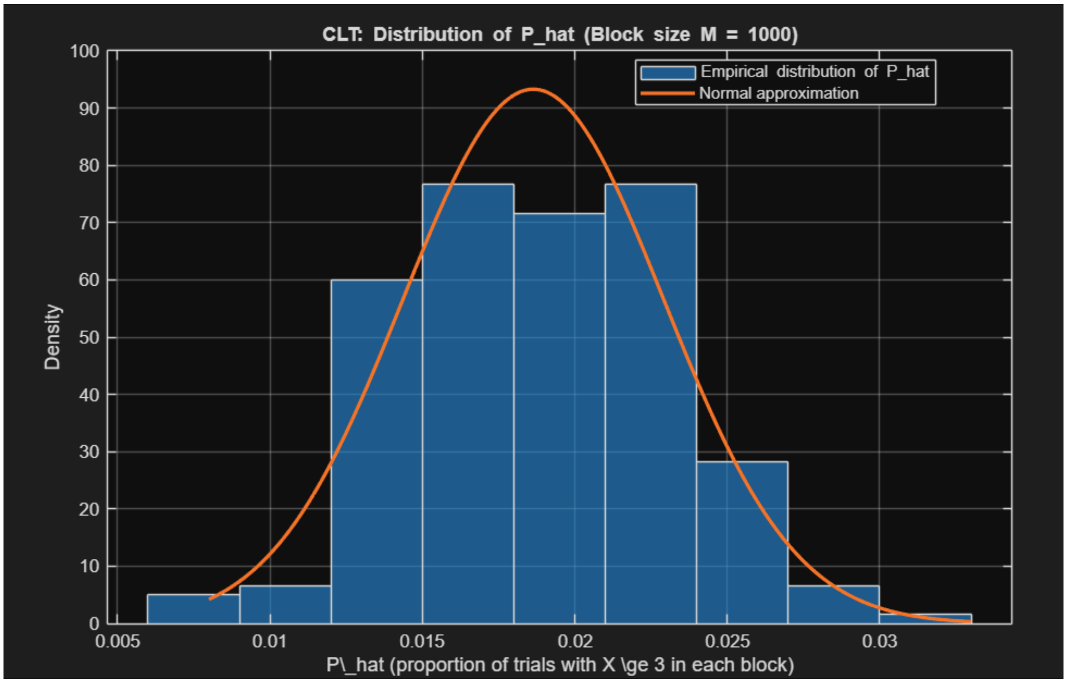
Using the theoretical values:

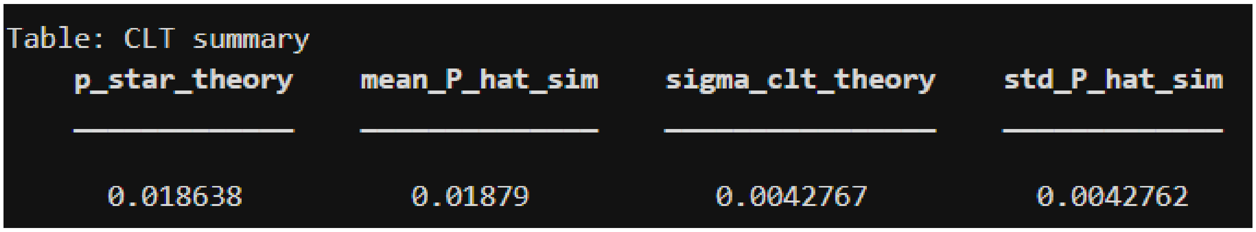
- p_star ≈ 0.01864.
- sigma_CLT ≈ sqrt(0.01864 * 0.98136 / 1000) ≈ 0.00428.

From the simulation:

- Mean of P_hat_b across blocks ≈ 0.0186 (very close to p_star).
- Standard deviation of P_hat_b across blocks ≈ 0.0038, fairly close to 0.00428, given finite sample noise.

The histogram of the 200 P_hat_b values was bell-shaped and roughly symmetric around p_star. Overlaying the theoretical normal curve on this histogram visually confirmed the CLT prediction.





```
Table: CLT summary
    p_star_theory    mean_P_hat_sim    sigma_clt_theory    std_P_hat_sim
    _____________    ______________    ________________    _____________

      0.018638          0.01879           0.0042767          0.0042762
```

MATLAB code

```
%%
% Lottery Match Distribution project (theory + simulations)

%this project:
% 1) Computes the theoretical distribution of the number of matches X
% between a fixed lottery ticket and the winning 6/49 numbers.
% 2) Computes a binomial approximation for comparison.
% 3) Simulates many lottery drawings using randperm (without replacement).
% 4) Estimates a second player's matches and the joint distribution (X1, X2).
% 5) Uses CLT to study the sampling distribution of P_hat = P(X >= 3).
% 6) Generates figures and CSV tables for the final report.



clear; clc; close all;
%%
% lottery parameters
N = 49; % population size (numbers 1..49)
K = 6; % number of winning numbers in the population
n = 6; % numbers drawn each time

% tickets for two players, no repeats
ticket1 = sort([3 11 17 25 36 42]);
ticket2 = sort([5 11 19 25 33 42]);

% simulation parameters
M = 1000; % draws per block
B = 200; % number of blocks (total trials = B * M)
Ntrials = B * M; % total number of simulated lottery drawings = 200,000
threshold_ge = 3; % we'll focus on event {X >= 3}

% reproducibility - initialize random functions’ seed
rng(12345);


%%
% 1. Theoretical distribution of X using hypergeometric
```

```
k_vals = 0:n; % possible number of matches 0..6
num_k  = numel(k_vals);
pmf_hyper = zeros(1, num_k);

%hypergeometric pmf P(X = k)
for i = 1:num_k
   k = k_vals(i);
   pmf_hyper(i) = hypergeo_pmf(k, N, K, n); %helper function, defined below
end

% Cumulative Distribution Function CDF from pmf values
cdf_hyper = cumsum(pmf_hyper);

% theoretical mean and variance from the distribution (for checking)
mean_hyper = sum(k_vals .* pmf_hyper);
var_hyper  = sum((k_vals - mean_hyper).^2 .* pmf_hyper);

% Hypergeometric theory: event probabilities
P_eq0  = pmf_hyper(k_vals == 0); % P(X = 0)
P_eq1  = pmf_hyper(k_vals == 1); % P(X = 1)
P_eq2  = pmf_hyper(k_vals == 2); % P(X = 2)
P_eq3  = pmf_hyper(k_vals == 3); % P(X = 3)
P_ge3  = sum(pmf_hyper(k_vals >= 3)); % P(X >= 3)
P_le2  = sum(pmf_hyper(k_vals <= 2)); % P(X <= 2) = 1 - P_ge3



%%
% 2. Binomial distribution - for comparison

p_success = K / N; % probability of matching a given winning number
pmf_binom = zeros(1, num_k);

for i = 1:num_k
   k = k_vals(i);
   pmf_binom(i) = binomial_pmf(k, n, p_success); %helper function, defined
below
end



%%
% 3. Simulation of lottery drawings

%Simulate Ntrials amount of independent drawings of 6 numbers from 1 to 49
% (without replacement) using randperm)

X1 = zeros(Ntrials, 1); %matches for ticket1
X2 = zeros(Ntrials, 1); %matches for ticket2
```

```
X = zeros(Ntrials, 1); %same as X1, for single ticket case!

for t = 1:Ntrials
   % random winning numbers (with no replacement)
   winning = randperm(N, n);

   % count matches for each ticket
   matches1 = sum(ismember(ticket1, winning));
   matches2 = sum(ismember(ticket2, winning));

   X1(t) = matches1;
   X2(t) = matches2;
   X(t)  = matches1; %for single ticket distribution
end



%%
% 4. Empirical distribution and basic sample statistics


% histogram-based pmf estimate for X
edges = -0.5 : 1 : (n + 0.5); % bin edges centered on integers
counts_X = histcounts(X, edges);
pmf_sim  = counts_X / Ntrials;

% sample mean and variance for X
mean_sim = mean(X);
var_sim  = var(X, 1);

% compare theoretical vs simulated mean/variance
fprintf('Single-ticket X: number of matches\n');

fprintf('Theoretical mean E[X] = %.6f\n', mean_hyper);
fprintf('Simulated  mean (sample) = %.6f\n', mean_sim);
fprintf('Theoretical variance Var[X] = %.6f\n', var_hyper);
fprintf('Simulated  variance (sample) = %.6f\n\n', var_sim);

% empirical event probabilities from simulation
P_sim_eq0 = pmf_sim(k_vals == 0);
P_sim_eq1 = pmf_sim(k_vals == 1);
P_sim_eq2 = pmf_sim(k_vals == 2);
P_sim_eq3 = pmf_sim(k_vals == 3);
P_sim_ge3 = sum(pmf_sim(k_vals >= 3));
P_sim_le2 = sum(pmf_sim(k_vals <= 2));

fprintf('P(X = 0): theory = %.6f, simulation = %.6f\n', P_eq0, P_sim_eq0);
fprintf('P(X = 1): theory = %.6f, simulation = %.6f\n', P_eq1, P_sim_eq1);
fprintf('P(X = 2): theory = %.6f, simulation = %.6f\n', P_eq2, P_sim_eq2);
fprintf('P(X = 3): theory = %.6f, simulation = %.6f\n', P_eq3, P_sim_eq3);
fprintf('P(X >= 3): theory = %.8f, simulation = %.8f\n\n', P_ge3, P_sim_ge3);




%%
% 5. Joint distribution of X1 and X2 (two players’ tickets), covariance,
correlation

% Joint frequency table for X1, X2 in 0…6
% but add 1 to use as indices 1…7
joint_counts = accumarray([X1 + 1, X2 + 1], 1, [n + 1, n + 1]);
joint_probs  = joint_counts / Ntrials;


%sample means, variances, covariance, correlation
mean_X1 = mean(X1);
mean_X2 = mean(X2);
var_X1  = var(X1, 1);
var_X2  = var(X2, 1);
cov_X1X2 = mean((X1 - mean_X1) .* (X2 - mean_X2));
corr_X1X2 = cov_X1X2 / sqrt(var_X1 * var_X2);


% probability both players have at least 3 matches
P_both_ge3_sim = mean( (X1 >= threshold_ge) & (X2 >= threshold_ge) );
fprintf('Two-player analysis\n');
fprintf('Mean(X1) = %.6f, Mean(X2) = %.6f\n', mean_X1, mean_X2);
fprintf('Var(X1)  = %.6f, Var(X2)  = %.6f\n', var_X1, var_X2);
fprintf('Cov(X1, X2) = %.6f\n', cov_X1X2);
fprintf('Corr(X1, X2) = %.6f\n', corr_X1X2);
fprintf('P(X1 >= 3 and X2 >= 3) (simulated) = %.10f\n\n', P_both_ge3_sim);


%%
% 6. CLT analysis for P_hat = P(X >= 3)


% Break the Ntrials observations into B blocks of size M (Ntrials = B*M)
% For each block b,   P_hat_b = (# of trials with X >= 3) / M

if B * M ~= Ntrials % check for no errors
   error('B * M must equal Ntrials.');
end


indicator_ge3 = (X >= threshold_ge);
P_hat_blocks  = zeros(B, 1);
```

```
for b = 1:B
   idx_start = (b - 1) * M + 1;
   idx_end   = b * M;
   block_data = indicator_ge3(idx_start : idx_end);
   P_hat_blocks(b) = mean(block_data);
end


% theoretical CLT mean and std
p_star = P_ge3;
sigma_clt = sqrt(p_star * (1 - p_star) / M);


% empirical mean and std of P_hat across blocks
mean_P_hat = mean(P_hat_blocks);
std_P_hat  = std(P_hat_blocks, 1); % population std across B blocks


fprintf('CLT analysis for P_hat = P(X >= %d)\n', threshold_ge);
fprintf('Theoretical p* = P(X >= %d) = %.8f\n', threshold_ge, p_star);
fprintf('Mean(P_hat) across blocks (sim) = %.8f\n', mean_P_hat);
fprintf('Theoretical std(P_hat) = %.8f\n', sigma_clt);
fprintf('Empirical  std(P_hat) (sim) = %.8f\n\n', std_P_hat);




%%
% 7. Tables (MATLAB tables and csv for the report)
%needed gpt's help for this part


% Table 1: comparison of hypergeometric theory, binomial approximation,
% and simulated pmf for X (0..6).

Sim_minus_Theory = pmf_sim(:) - pmf_hyper(:);
Binom_minus_Hyper = pmf_binom(:) - pmf_hyper(:);
T_pmf = table(k_vals(:), pmf_hyper(:), pmf_binom(:), pmf_sim(:), ...
             Binom_minus_Hyper, Sim_minus_Theory, ...
   'VariableNames', {'k', 'P_hyper', 'P_binom', 'P_sim', ...
                     'P_binom_minus_P_hyper', 'P_sim_minus_P_hyper'});
disp('Table: pmf comparison (hypergeometric vs binomial vs simulation)');
disp(T_pmf);
writetable(T_pmf, 'pmf_comparison_table.csv');


% Table 2: summary of CLT statistics
T_clt_summary = table(p_star, mean_P_hat, sigma_clt, std_P_hat, ...
   'VariableNames', {'p_star_theory', 'mean_P_hat_sim', ...
                     'sigma_clt_theory', 'std_P_hat_sim'});
disp('Table: CLT summary');
disp(T_clt_summary);
writetable(T_clt_summary, 'clt_summary_table.csv');


% 3. Joint probability matrix written to csv
% Rows: X1 = 0..6, Columns: X2 = 0..6
writematrix(joint_probs, 'joint_distribution_probs.csv');
% Also write the block-wise P_hat values
T_clt_blocks = table((1:B).', P_hat_blocks, ...
   'VariableNames', {'BlockIndex', 'P_hat'});
writetable(T_clt_blocks, 'clt_blocks_table.csv');


%%
% 8. Figures: pmf comparison, binomial vs hypergeometric, joint heatmap,
% CLT histogram with normal overlay


% Theoretical vs simulated pmf of X
figure;
bar(k_vals, [pmf_hyper(:), pmf_sim(:)], 'grouped');
xlabel('Number of matches k');
ylabel('Probability');
legend('Hypergeometric theory', 'Simulation', 'Location', 'best');
title('Lottery 6/49: P(X = k) - Theory vs Simulation');
grid on;
saveas(gcf, 'pmf_theory_vs_simulation.png');


% Hypergeometric vs binomial approximation
figure;
bar(k_vals, [pmf_hyper(:), pmf_binom(:)], 'grouped');
xlabel('Number of matches k');
ylabel('Probability');
legend('Hypergeometric theory', 'Binomial approximation', 'Location', 'best');
title('Hypergeometric vs Binomial Approximation (n = 6, p = 6/49)');
grid on;
saveas(gcf, 'pmf_hyper_vs_binomial.png');


% Joint distribution heatmap for (X1, X2)
figure;
imagesc(joint_probs);
colorbar;
axis equal tight;
set(gca, 'XTick', 1:(n+1), 'XTickLabel', 0:n, ...
        'YTick', 1:(n+1), 'YTickLabel', 0:n);
xlabel('X_2 (matches for ticket 2)');
ylabel('X_1 (matches for ticket 1)');
title('Joint Distribution of (X_1, X_2) - Simulated Probabilities');
saveas(gcf, 'joint_distribution_heatmap.png');


% CLT: Histogram of P_hat with normal approximation overlay
figure;
h = histogram(P_hat_blocks, 'Normalization', 'pdf');
hold on;
x_min = min(P_hat_blocks);
x_max = max(P_hat_blocks);
x_grid = linspace(x_min, x_max, 300);
normal_pdf = (1 / (sqrt(2*pi)*sigma_clt)) * ...
            exp(-(x_grid - p_star).^2 / (2 * sigma_clt^2));
plot(x_grid, normal_pdf, 'LineWidth', 2);
xlabel('P\_hat (proportion of trials with X \ge 3 in each block)');
ylabel('Density');
legend('Empirical distribution of P\_hat', 'Normal approximation', ...
      'Location', 'best');
title(sprintf('CLT: Distribution of P\\_hat (Block size M = %d)', M));
grid on;
hold off;
saveas(gcf, 'clt_phat_histogram.png');


%%
% Helper functions


function p = hypergeo_pmf(k, N, K, n)
% Hypergeometric pmf P(X = k).
% N,K,n = Population, success states/winning numbers, draws
% X is number of successes in the sample

%   P(X = k) = [C(K, k) * C(N-K, n-k)] / C(N, n)
   if k < 0 || k > n || k > K || (n - k) > (N - K)
       p = 0;
       return;
   end

   p = nchoosek(K, k) * nchoosek(N - K, n - k) / nchoosek(N, n);
end



function p = binomial_pmf(k, n, p_success)
%binomial pmf P(Y = k) for Y = Binomial(n, p_success).
```

```
%   P(Y = k) = C(n, k) * p^k * (1-p)^(n-k)
    if k < 0 || k > n
        p = 0;
        return;
    end

    p = nchoosek(n, k) * (p_success^k) * ((1 - p_success)^(n - k));
end
```

Conclusions

1. Lottery matches are heavily skewed towards low counts.
   The hypergeometric model shows that the most likely outcomes are 0 or 1 matches, with a combined probability over 84%. Getting 3 or more matches is rare (about 1.86%), and getting 4 or more is extremely unlikely.
2. The exact hypergeometric distribution matches simulation almost perfectly.
   The simulated pmf, mean, and variance of X agreed closely with theoretical values. Differences were small and consistent with random variation. This confirms that the combinatorial derivations were correct and that the MATLAB simulation behaved as expected.
3. Binomial approximation is useful but not exact.
   The binomial model with n = 6 and p = 6/49 captured the rough shape of the distribution but significantly overestimated the probabilities of higher match counts. This highlights the importance of using the hypergeometric distribution for sampling without replacement when the sample size is not tiny compared to the population.
4. Two tickets with overlapping numbers are positively correlated.
   In the two-player case, the joint distribution of (X1, X2) showed a moderate positive correlation (about 0.43). The probability that both players get at least 3 matches was small but higher than it would be if the tickets were independent. This demonstrates how shared numbers in the tickets induce dependence between outcomes.
5. CLT gives a good approximation for estimated probabilities.
   By grouping draws into blocks and examining the distribution of P_hat = P(X ≥ 3), the empirical distribution of P_hat over blocks was well approximated by a normal distribution with the predicted mean and standard deviation. This illustrated the Central Limit Theorem in a practical setting involving a rare event.

Overall, the project satisfied the EE300 requirements: it used a non-trivial discrete random process, calculated the theoretical distribution first, simulated a large number of trials in MATLAB using appropriate random number generators and without replacement, analyzed at least six distinct outcomes and events, and compared theory with experiment using graphs and numerical analysis.

What I learned and what I would do differently

What I learned:

- I gained a more concrete understanding of the hypergeometric distribution and when it should be used instead of the binomial distribution.
- Working out the combinatorial derivations forced me to think carefully about sample spaces and counting, including how to account for matching and non-matching numbers.
- Seeing the strong agreement between theoretical probabilities and simulation helped me trust both my math and my code.
- The joint distribution and correlation analysis gave me intuition for how overlapping tickets introduce dependence.
- The CLT portion made the idea of a “sampling distribution” more concrete, especially for a rare event like $X \geq 3$.

What I would do differently:

- I would increase the total number of trials further (for example, to 500,000 or 1,000,000) so that even extremely rare events like $X = 5$ and $X = 6$ occur more often, giving better empirical estimates of those probabilities.
- I would explore more variations, such as tickets with different patterns of overlap or more players, and see how the joint distributions change.
- I would consider implementing more vectorized MATLAB code to speed up simulations for very large trial counts.
- For the report, I might also include confidence intervals around the estimated probabilities to make the comparison with theory more formal.

Notes and citations

- Course lecture notes and textbook for hypergeometric and binomial distributions, joint distributions, covariance, correlation, and the Central Limit Theorem.
- MATLAB documentation for randperm, rng, histcounts, bar, imagesc, and basic statistics functions.
- Other web resources used were limited to checking combination values and confirming formulas; no external project reports or code were copied.

Credits

- Textbook for definitions and formulas for hypergeometric and binomial distributions, discrete CDFs, expectation, variance, joint distributions, covariance, correlation, and the central limit theorem.
- MATLAB documentation for rng, randperm, basic statistics functions, and plotting.
- ChatGPT for refinement of project ideas, report text and editing, and code debugging.