5.13.62

M Chanakya Srinivas- EE25BTECH11036

Problem Statement

Question:

How many 3×3 matrices M, with entries from the set $\{0, 1, 2\}$, satisfy:

$$\text{tr}(M^\top M) = 5 \quad (1)$$

Matrix Structure

Let the matrix $M \in \mathbb{R}^{3\times 3}$ be:

$$M = \begin{pmatrix} \mathbf{c_1} & \mathbf{c_2} & \mathbf{c_3} \end{pmatrix} \tag{2}$$

where each column vector is:

$$\mathbf{c_j} = \begin{pmatrix} c_{1j} \\ c_{2j} \\ c_{3j} \end{pmatrix}, \quad j = 1, 2, 3 \tag{3}$$

Each $c_{ij} \in \{0, 1, 2\}$, so total possible vectors:

$$3^3 = 27 \tag{4}$$

Computing $M^\top M$

The product:

$$M^\top M = \begin{pmatrix} \mathbf{c_1}^\top \mathbf{c_1} & \mathbf{c_1}^\top \mathbf{c_2} & \mathbf{c_1}^\top \mathbf{c_3} \\ \mathbf{c_2}^\top \mathbf{c_1} & \mathbf{c_2}^\top \mathbf{c_2} & \mathbf{c_2}^\top \mathbf{c_3} \\ \mathbf{c_3}^\top \mathbf{c_1} & \mathbf{c_3}^\top \mathbf{c_2} & \mathbf{c_3}^\top \mathbf{c_3} \end{pmatrix} \tag{5}$$

Trace of the matrix is:

$$\begin{aligned} \mathrm{tr}(M^\top M) &= \mathbf{c_1}^\top \mathbf{c_1} + \mathbf{c_2}^\top \mathbf{c_2} + \mathbf{c_3}^\top \mathbf{c_3} && (6) \\ &= \|\mathbf{c_1}\|^2 + \|\mathbf{c_2}\|^2 + \|\mathbf{c_3}\|^2 && (7) \end{aligned}$$

Norm Constraint

Let:

$$n_1 = \|\mathbf{c_1}\|^2, \quad n_2 = \|\mathbf{c_2}\|^2, \quad n_3 = \|\mathbf{c_3}\|^2 \tag{8}$$

We want:

$$n_1 + n_2 + n_3 = 5 \tag{9}$$

Norm Counts for Vectors

Define $N(n)$ as number of vectors $\mathbf{v} \in \{0,1,2\}^3$ with squared norm n. Then:

$$
\begin{align}
N(0) &= 1 \quad \left(\text{only } \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}\right) \tag{10} \\
N(1) &= 3 \quad (\text{one entry } 1, \text{ others } 0) \tag{11} \\
N(2) &= 3 \quad (\text{one entry } 2, \text{ others } 0) \tag{12} \\
N(3) &= 6 \tag{13} \\
N(4) &= 3 \tag{14} \\
N(5) &= 6 \tag{15}
\end{align}
$$

Counting Valid Matrices

We count all ordered triples $(\mathbf{c_1}, \mathbf{c_2}, \mathbf{c_3})$ such that:

$$n_1 + n_2 + n_3 = 5 \tag{16}$$

Total count is:

$$\text{Total} = \sum_{\substack{n_1+n_2+n_3=5 \\ 0 \le n_i \le 5}} N(n_1) \cdot N(n_2) \cdot N(n_3) \tag{17}$$

Evaluating this sum gives:

$$\boxed{198} \tag{18}$$

Final Answer

$$\boxed{\text{Number of matrices } M = 198} \tag{19}$$

C code

```c
#include <stdint.h>
int get_count(void) {
    int count = 0;
    // There are 9 entries; treat them as base-3 digits 0,1,2
    for (int mask = 0; mask < 19683; ++mask) { // 3^9 = 19683
        int tmp = mask;
        int sumsq = 0;
        for (int i = 0; i < 9; ++i) {
            int digit = tmp % 3; // 0,1,2
            tmp /= 3;
            sumsq += digit*digit;
            if (sumsq > 5) break; // small optimization
        }
        if (sumsq == 5) ++count;
    }
    return count;
}
```

Python code through shared output

```
import ctypes
import os

# Adjust path if needed
libpath = os.path.abspath(./libcount.so)
lib = ctypes.CDLL(libpath)

lib.get_count.restype = ctypes.c_int

result = lib.get_count()
print(Number of 3x3 matrices with trace(M^T M)=5:, result)
```

Only Python code

```
import numpy as np
from itertools import product

def count_matrices():
    # Step 1: compute squared norm multiplicities for column
        vectors in {0,1,2}^3
    cols = np.array(list(product((0,1,2), repeat=3)))
    norms = np.sum(cols**2, axis=1)
    unique, counts = np.unique(norms, return_counts=True)
    N = dict(zip(unique, counts)) # N[n] = multiplicity of norm n
```

Only Python code

```
    # Step 2: sum over all triples of norms that add to 5
    total = 0
    for n1, c1 in N.items():
        for n2, c2 in N.items():
            for n3, c3 in N.items():
                if n1 + n2 + n3 == 5:
                    total += c1 * c2 * c3
    return total, N

if __name__ == __main__:
    total, N = count_matrices()
    print(Single-column squared norm multiplicities N(n):)
    for n in sorted(N):
        print(f n={n}: {N[n]} ways)
    print(\nTotal matrices =, total)
```