Homework on Probability Spaces

Tokyo Data Science

Due on Sunday, October 25, 2020

Submission

Use the following form to submit your homework:

https://airtable.com/shrRsw4y1y5riMeLS

When you enter a correct answer into the form, it will immediately indicate that the answer is correct. If you believe that you have the correct answer but the form does not recognize it as such, please contact me. If you have submitted the form and later you found solutions to questions you were not able to answer previously, you can submit this form again.

Task 1

Consider random images with 100×100 pixels that can be either black or white. How many elements does the associated sample space have? How does this compare to the number of atoms in the observable universe?

Task 2

A probability space consists of three ingredients: an outcome space Ω, a class of events $\mathcal{A}$, and a probability measure P. We require that the class of events is a so-called σ-field (sigma-field). A σ-field $\mathcal{A}$ is a collection of events that satisfies the following three conditions:

(1) It contains the empty set (i.e. $\emptyset \in \mathcal{A}$).

(2) For every event A, it also contains its complement A^c (i.e. the set that contains all elements of Ω except for those in A).

(3) For any series of events $A_1, A_2, A_3, ...$, it contains also its union $\cup_{i=1}^{\infty} A_i$.

Stated differently, condition (2) requires that the σ-field is closed under complementation, and condition (3) requires that the σ-field is closed under countable union.

Consider one roll of a die, with sample space $\{1, 2, 3, 4, 5, 6\}$. What is the smallest σ-field that contains the event $\{1, 3, 5\}$? How many elements does it have? What is the smallest σ-field that contains the events $\{1, 2\}$ and $\{1, 2, 3, 4\}$? How many elements does it have?

Task 3 (optional, for those getting started with Python or probability)

Using Numpy in Python, generate 10000 random numbers drawn from the uniform distribution on the interval $[0, 1]$, organized into a Numpy array x. Then create a Numpy array y containing the entries of array x squared. Calculate the mean value in the array y and report it with a precision of 4 decimal places. If you already have the relevant knowledge, what is the exact mean (expectation) of the square of a random variable uniformly distributed on the interval $[0, 1]$? Report it with a precision of 4 decimal places.

.

Note: Strictly speaking, in Numpy we are using pseudorandom numbers, not random numbers.