Міністерство освіти і науки України
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «КИЄВО-МОГИЛЯНСЬКА АКАДЕМІЯ»
Кафедра математики факультету інформатики

NK-моделі Кауффмана

Курсова робота

за спеціальністю „Прикладна математика ” 113

Керівник курсової роботи

д.т.н., доц. Чорней Р.К.

____

(підпис)

“____” _________ 2020 р.

Виконав

студент 4 курсу

факультету інформатики

Крошин О. А.

Міністерство освіти і науки України
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «КИЄВО-МОГИЛЯНСЬКА АКАДЕМІЯ»
Кафедра інформатики факультету інформатики

ЗАТВЕРДЖУЮ
Зав. кафедри математики,
проф., д.ф.-м.н.
____________Б. В. Олійник
(підпис)
„____” ________________2019 р.

ІНДИВІДУАЛЬНЕ ЗАВДАННЯ
на курсову роботу

студенту Крошину О. А. факультету інформатики 4-го курсу БП
ТЕМА NK-моделі Кауффмана
Зміст курсової роботи:

1. Індивідуальне завдання
2. Календарний план
3. Анотація
4. Вступ
5. РОЗДІЛ 1: Загальна інформація про NK-моделі Кауффмана
6. РОЗДІЛ 2: Аналіз властивостей NK-моделей Кауффмана з різними K
7. Висновки
8. Список використаних джерел

Дата видачі „___” _________ 2019 р. Керівник ________________
(підпис)
Завдання отримав ___________
(підпис)

Календарний план виконання роботи:

№ п/п	Назва етапу курсової роботи	Термін виконання етапу	Примітка
1.	Отримання теми курсової роботи.	10.10.2019	
2.	Огляд задачі	11.11.2019	
3.	Побудова програми-симулятора	01.02.2020	
4.	Написання першого розділу	01.03.2020	
5.	Проведення досліджень	15.03.2020	
6.	Написання другого розділу	01.04.2020	
7.	Корегування роботи згідно із зауваженнями керівника	10.04.2020	

Студенту Крошину О.А.

Керівник Чорней Р.К.

“______”_______________

Table of contents

Abstract

A given work focuses on the NK fitness models, focuses on classical NK models of Boolean networks dynamics, gives intuition of its basic properties and describes most useful ways to represent it. The paper is also focused on a fitness landscape and its corresponding NK model, concepts of ruggedness and smoothness. It also covers a concept of NK-models stability and researches an impact of internal and external parameters of a model (N, K, ruggedness of a landscape) on its stability.

Experiments were implemented with Python 3.8 using libraries Numpy, matplotlib etc.

Introduction

Proposed by S. Kauffmann, the NK simulation model was originally intended to serve the purposes of theoretical implications in evolutionary biology. With its capabilities for capturing network dynamics, it raised high interest in different fields. Its methodology is traditionally split between Boolean network modelling, which is studying proper boolean networks dynamics, and a problem of fitness-driven network evolution, dedicated to systems evolution based on search for optimizing its fitness value through a so-called fitness landscape. The former is traditionally labelled as NK-BN, and the latter is referred to as NK-FL. Both approaches were linked by Kauffman himself, when he defined fitness landscape as a Boolean network space.

Splitting the two NK models is justified by disciplinary interests: while NK-BN is mostly researched by authors in fields of biology, mathematics and computer science (e.g. viewing NK-model as an NP-complete problem (Weinberger, 2014), NK-FL attracts lots of attention in fields of project management (e.g. dealing with uncertainty, Levinthal and Workiewicz, 2017), business (modelling product launches, Sommer et al., 2004) and economics (forming governmental industrial policies, Li and Csaszar, 2018)

The aim of this paper is to review both types of NK-model, discuss its limitations and give intuition of its landscape ruggedness based on tuning K and conducting additional experiments on model stability.

Paper consists of three parts, where the first focuses on basic intuition around NK-model and its representation, the second on proper experiments and the third on interpreting their results.

1. The NK models, Boolean network dynamics and Fitness Landscape

1.1.The NK models.

The problem of capturing networks dynamics has a long history of generating new proposals and approaches. The computational limitations in determining all states of complex systems and transition between their phases led to creation of simpler models, which despite growing in time with an increasing number of parameters, let a researcher explore only some part of its states, providing some intuition of generalizing it further. This paper is dedicated to one of the fundamental approaches to the given problem.

The NK model was originally defined by Kauffman as “meant to be applied to systems of many, N, parts, where the functional contribution of each part depends upon the state, among A alternatives, of that part, and is epistatically affected by average of K other parts” [2].

This particular quote lets us broadly define a component of NK-model as a sequence of N elements where each element takes one of A possible values and is calculated as function of other K elements, also known as a fitness function.

K measures the richness of the interconnectivity of the parts of the given system, and the main problem behind NK-model in general is the following: does tuning just two parameters, a fitness function and its size K, change its behavior fundamentally? Do different values of K form different “regimes” with completely differently dynamics? Are there phase transitions between the regimes? Short answer: yes, it does, which will be demonstrated below.

1.2. The NK-BN model

The standard version of NK-model, known as NK-BN, is heavily tied to a theory of random Boolean networks. NK-BN model (or system) methodology is built upon defining a model as topologically invariant, meaning that both its connections distribution and its size are defined stochastically as relationship of its nodes based on invariant interaction rules (fitness functions) and that a model does not change neither in size nor in its connections distribution. Its main principles may be stated as follows:

1) Every node has a number of activation states (usually, two, 0 or 1, active or inactive)
2) Edges between node are combined by Boolean functions, sometimes called activation rules.
3) Every node receiving edges is determined as (in)active as output of a combination of edges and impacts a distribution of states at the whole network.
4) Given a set of Boolean functions, we can determine all the states in of a network until it reaches a final stable state.
5) A final stable state is called attractor, and may be either a single state or a series of consecutive and repetitive states which form a cycle, known as a cyclic attractor.
6) Main property of an attractor is that its' transition between states don't affect distributions of nodes' activation states.
7) Every initial state starts a specific dynamics (trajectory), which can coincide with another trajectory. Finally, a set of all trajectories forms a network state space (combinatorial phase space).

For every initial state always exists at least one attractor. When a network reaches attractor states, its' dynamics doesn't stop, but its nodes distribution remains constant.

For a BN model of size K and unique activation states T we can calculate total number of its possible states as 2^N, number of possible transitions between states as 2^{2^N} and number of possible activation rules as 2^{T^K}. A notion of how crucial K is may be illustrated with a following table:

N	**K**	**T**	**Number of states**	**Number of transitions**	**Number of activation rules**
5	1	2	32	4294967296	2
5	2	2	32	4294967296	16
5	3	2	32	4294967296	256
5	4	2	32	4294967296	65536
…	…	…	…	…	…
N	K	T	2^N	2^{2^N}	2^{T^K}

Table 1 –number of activation rules depending on K

It may be seen that with the increase of K grows the computational resource to reach the stable state of the system. Kauffman demonstrated that the system becomes more chaotic. Note that in this paper we assume that T=2 unless other value of T is specified.

1.3 NK FL model. Fitness function. Landscape ruggedness

Evolutionary biology is rich for constructs. One of such constructs is called a fitness landscape. Proposed by S. Wright in 1932, it was created with an aim of exploring success rates of a particular gene or genotype through time. It is heavily based on a concept of fitness function, a class of functions used as a metric of a gene's similarity to a set goal (or to gene's replication rate). From a biological point of view, the fitness landscape may be formed by the set of all necessary genotypes, some information about similarity between them and their specific fitness values. Main purpose of fitness landscapes in biology is evaluating proposed sets of previously fitted best parameters for a certain genetic problem.

Abstractly, given a certain alphabet P, a set of all possible strings of length N we can form a combinatorial phase space, where each element defines its own value, called fitness. Every component of a string N depends in average on K previous components. With a defined distance metric between strings, the resulting structure is called a landscape (or fitness landscape) and may be applied to a wide range of tasks via imitating mechanisms of evolutionary biology. Main principles of NK-FL model:

1) There are N components describing some characteristics. An average amount of connections between components is determined by K
2) Fitness value of a characteristic is calculated as a mean of corresponding fitness elements.

Fitness landscapes consist of local peaks – points with paths leading to lower fitness, and valleys – points, leading fitness values upwards.

Fitness landscape with low number of peaks and valleys is called smooth, and the one with high amount of them is called rugged.

1.4 Model representation

To give an intuition of the concepts represented above we'll finish this chapter by a review of different methods of representing an NK model and providing some small examples of both NK-BN and NK-FL. There are several ways of representing and NK model.

Firstly, the NK-model may be represented as an interaction matrix I of size NxN, where $\begin{cases} I[i,j] = 1 \text{ if I[i] } depends\ on\ I[j] \\ I[i,j] = 0, otherwise \end{cases}$

Example: $N = 4, K = 2; I = \begin{pmatrix} 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \end{pmatrix}$

Secondly, the NK-model may be represented as a directed graph, where each of N elements is represented as vertex, and an edge shows whether node X depends on node Y.

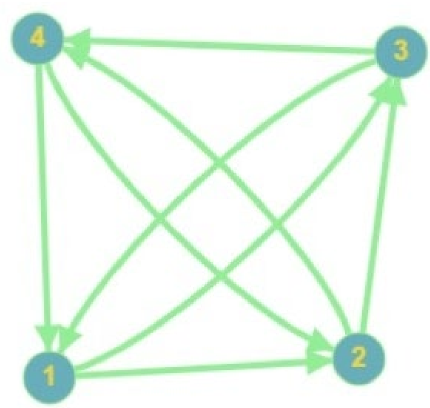

Figure 2 – a model with N=4, K=2

Alternatively, the NK-model may be represented as a hypercube, where each node equals the value of a fitness function from K arguments. An actor will determine whether to stay at a current peak (if fitness value of a current node is higher than of its neighbors), or to choose between K options. The main problem of that approach is its dependency on an initial state, with a high risk of getting stuck in a local optimum. An example below will be given for N=3, however, for N>3 hypercube should be seen as a graph with N neighbors for each node.

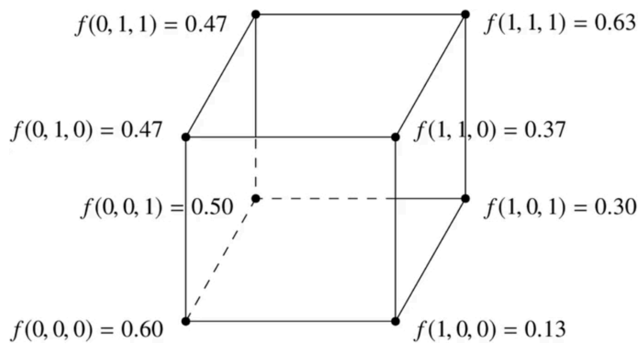

Figure 3 – hypercube for a model with N=4, K=2

Example: Looking for an attractor

For the NK-BN (N=3, K=2). and Boolean functions AND, OR, XOR, we will generate all possible initial states and determine steps to find a single (or cyclic) attractor. Keep in mind for small values of K the estimate of length of trajectory ($\sqrt{N}$) [6, p.4]

Initial state	**AND**	**OR**	**XOR**
	{transitions}; number of steps to reach attractor		
000	{000 → **000**};0	{000 → **000**};0	{000 → **000**};0
001	{001 → **000**};1	{001 → **111**};1	{001 → **101**};1
010	{010 → **000**};1	{010 → **111**};1	{010 → **110**};1
011	{011 → **111**};1	{011 → **111**};1	{011 → **011**};0
100	{100 → **000**};1	{100 → **000**};1	{100 → **000**};1
101	{101 → **000**};1	{101 → **111**};1	{101 → **101**};0
110	{110 → **000**};1	{110 → **111**};1	{110 → **110**};0
111	{111 → **111**}; 0	{111 → **111**}; 0	{111 → **011**}; 1

Table 2 –steps to reach an attractor state (cycle), N=3; attractors are in bold

As may be noticed, all the initial states reached an attractor in no more than one step. Specifically, among 24 unique tuples (initial state, boolean function) 8 tuples had the initial state and the attractor state being the same (among them three “000” and two “111”), and distribution of boolean functions is the following: “{AND:2, OR:2, XOR:4}”. The aforementioned trajectories of length 0 are called garden-of-Eden states.

Let’s have a look at a similar experiment with N=4:

Initial state	**AND**	**OR**	**XOR**
	{transitions}; number of steps to reach attractor		
0000	{0000 → **0000**};0	{0000 → **0000**};0	{0000 → **0000**};0
0001	{0001 → **0000**};1	{0001 → **1111**};1	{0001 → **1011** → **0110** → **1101** → 1011};3
0010	{0010 → **0000**};1	{0010 → **1111**};1	{0010 → **1101** → **1011** → **0110** → 1101};3
0011	{0011 → **1111**};1	{0011 → **1111**};1	{0011 → **0110** → **1101** → **1011** → 0110};3
0100	{0100 → **0000**};1	{0100 → **0000**};1	{0100 → **0000**};1
0101	{0101 → **0000**};1	{0101 → **1111**};1	{0101 → **1011** → **0110** → **1101** → 1011};3
0110	{0110 → **0000**};1	{0110 → **1111**};1	{**0110** → **1101** → **1011** → 0110};2
0111	{0111 → **1111**}; 1	{0111 → **1111**}; 1	{0111 → **0110** → **1101** → **1011** → 0110}; 3
1000	{1000 → **0000**};0	{1000 → **0000**};0	{1000 → **0000**};0
1001	{1001 → **0000**};1	{1001 → **1111**};1	{1001 → **1011** → **0110** → **1101** → 1011};1
1010	{1010 → **0000**};1	{1010 → **1111**};1	{1010 → **1101** → **1011** → **0110** → 1101};3
1011	{1011 → **1111**};1	{1011 → **1111**};1	{**1011** → **0110** → **1101** → 1011};2
1100	{1100 → **0000**};1	{1100 → **0000**};1	{1100 → **0000**};1
1101	{1101 → **0000**};1	{1101 → **1111**};1	{**1101** → **1011** → **0110** → 1101};2
1110	{1110 → **0000**};1	{1110 → **1111**};1	{1110 → **1101** → **1011** → **0110** → 1101};3
1111	{1111 → **1111**}; 0	{1111 → **1111**}; 0	{**1111** → **0110** → **1101** → **1011** → 0110};3

Table 3 –reaching an attractor state (cycle); N=4, (cyclic)attractors are in bold

Among 48 unique pairs of initial states and boolean functions the distribution of trajectory length L is the following: "{0: 8, 1: 29, 2: 3, 3: 8}", with all trajectories of $L > 2$ being produced by XOR. the distribution of boolean functions is the following: "{AND: {0:3,1:13}, OR: {0:3,1:13}, XOR: {0:2,1:3,2:3,3:8}}". Note that in actual modelling the tautologies (e.g. "000") will be automatically removed, as they don't hold any particular sense.

2. Tuning the NK Models. On the edge of chaos

2.1. The ruggedness of a landscape based on K

As it was mentioned earlier, value of K determines whether system is stable, chaotic or in some intermediate state. However, along a question of chaoticity may be raised a list of other questions, such as:

1) What number of local optima exists in a landscape? What distributions are they described with?
2) What are the uphill lengths of trajectories? How fast is decreasing the number of neighbors with higher fitness?
3) What quantity of alternative optima are available from a given initial state? What is the chance of hopelessly getting stuck in a local peak?
4) What portion of initial states can reach (local) optima?
5) How can be defined a fitness difference between local and global optima?

The extensive search made by Kauffman [2] shows that the in general these characteristics don't rely on the nature of the fitness landscape, but rather only by influence of N and K. For a fitness landscape, increasing N gives an agent higher range of choices, reducing number of possible local optima to get stuck. Thus, increasing N makes a landscape smoother.

As of tuning K, we should formally start with specifying the state with K=0. Elements of such network don't interact in any way, thus, the system in general ceases to exist.

Case with K=1 makes the system very sparse, and if takes relatively long trajectories to reach a few local optima.

The NK-model with K=N-1 generates an absolutely random fitness landscape. Number of local optima is very high (on average, $\frac{2^N}{N+1}$), with a mean length of trajectory close to ln (N).

Finally, state with increasing N and K=N is defined by some researches as "the complexity catastrophe", and is associated with decline in mean fitness value of local optima. Fitness value of global optima, however, stays the same after reaching the mean of fitness space [1][2].

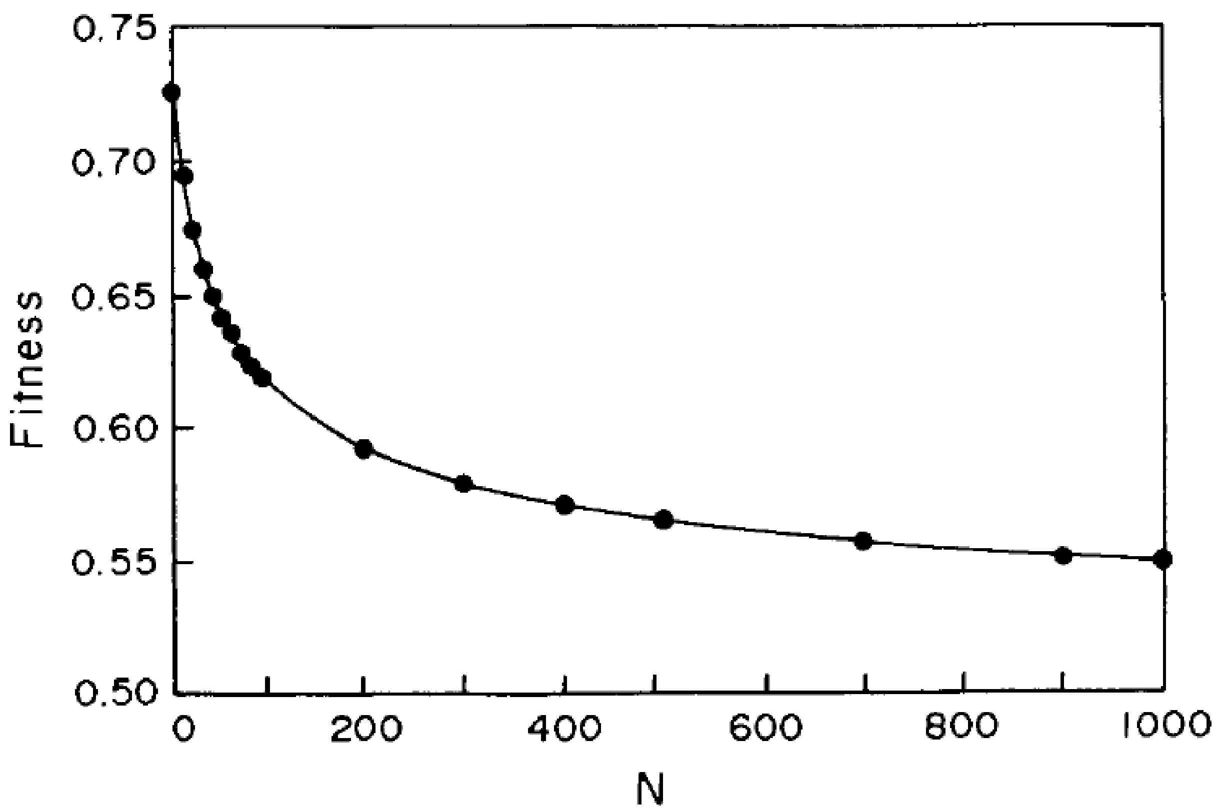

Figure 2 – a model with N=4, K=2

Overall, it is shown that increasing K from 1 to N-1 raises the number of peaks and valleys, consequently increasing the ruggedness of a landscape.

2.2. Stability of NK-model

In one of his earliest works [4], Kauffman was using randomly initialized networks to make an extensive search over a wide set of question. One particular question was to observe system's stability to infrequent noise. For an experiment, he used wide variety of nets of different N (between 15 and 2000) and up to 16 Boolean networks. The conclusion was that adding the noise provokes shift in cycles with low probabilities in range between 0.01 and 0.05 for K=2 and comparable with K=3. However, he was stopped from providing analysis on the highly-connected network of N~K by the expected length of cycle being close to 10^{150}. Later research introduced a notion of critical value of K, the aforementioned "edge of chaos". Given NK-FL model of Kauffman, we can define system as stable if $K < K_{critical}$, unstable if $K > K_{critical}$. Otherwise, we cannot say anything about its stability. [8]. For a random NK-BN, we can define its stability as follows:

Given p_i – a probability of transition to an active state, we define its sensitivity as $q_i = 2\frac{1}{\frac{1}{p_i}(1-p_i)^{-1}}$, and define the network is stable if $\max(\lambda_Q) < 1$, unstable if $\max(\lambda_Q) > 1$, and on the edge otherwise. $\lambda_Q$are the eigenvalues of $Q = qA$. A stands for an adjacency matrix of a system.

3. Practical research

3.1. Analyzing the number of peaks based on tuning

Let's analyze number of local peaks and mean value of fitness discussed in previous chapters. We'll use a system with $N = 8$, and $K = \{0,1,2,3,4,5,7,8\}$, to give an intuition about the controllable states $(K \leq 1)$, chaotic states $(K \geq 7)$. We will generate 1024 random interaction matrices and then calculate the distribution of peaks for the corresponding K.

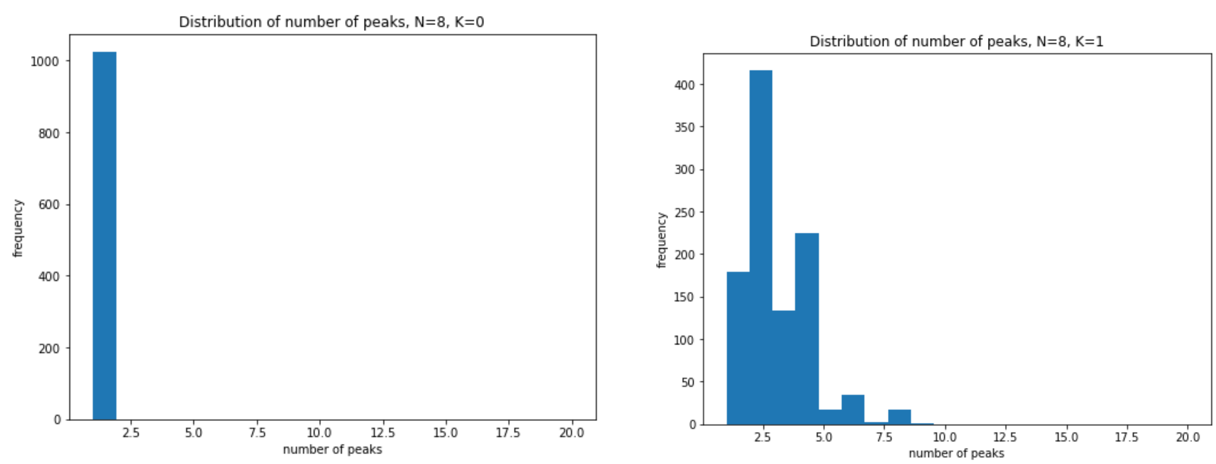

Figure 3 – NK with K=0 – "not a system", only one global peak; K=1 ~ 2.7 peaks sparsely connected data

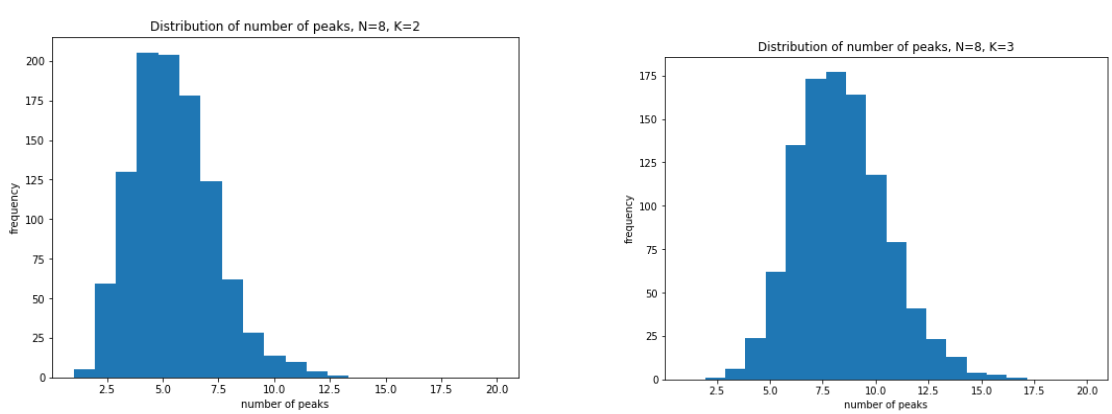

Figure 4 – NK the higher K, the bigger mean number of peaks

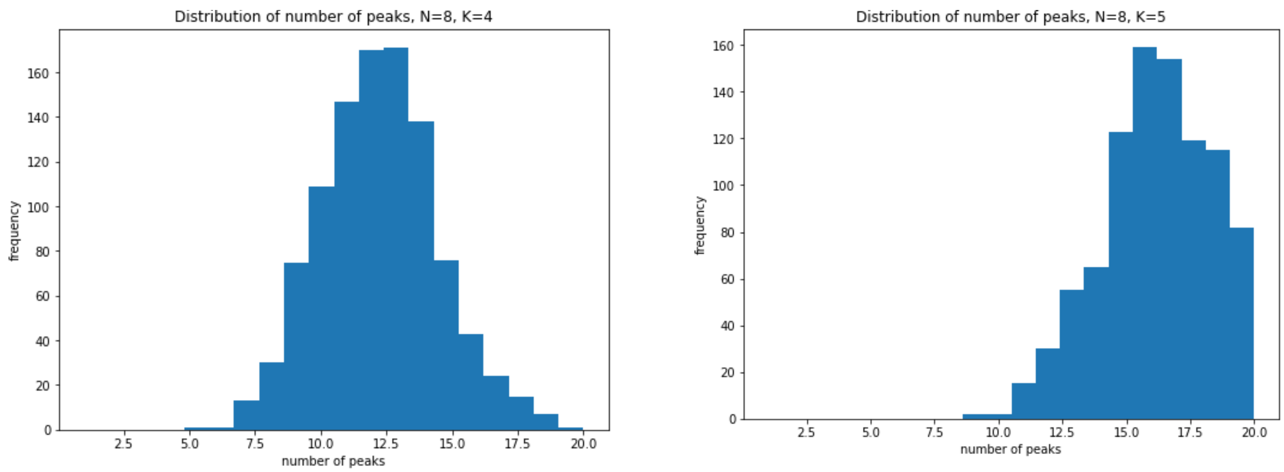

Figure 5 – K increased by 1, mean value increased by 25%

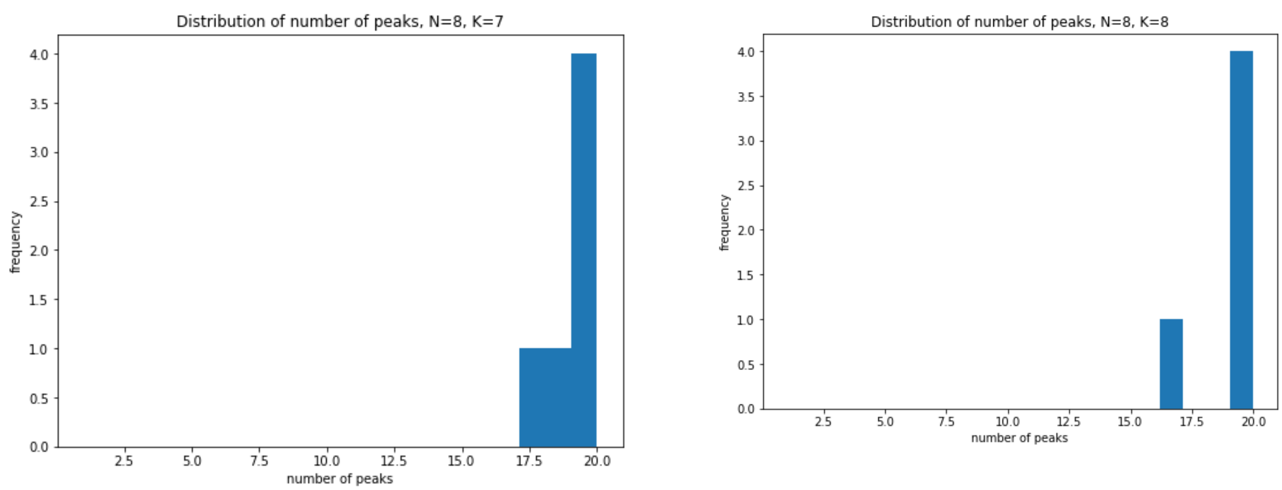

Figure 6 – with K close to N, mean value of peaks is ~ 28.4

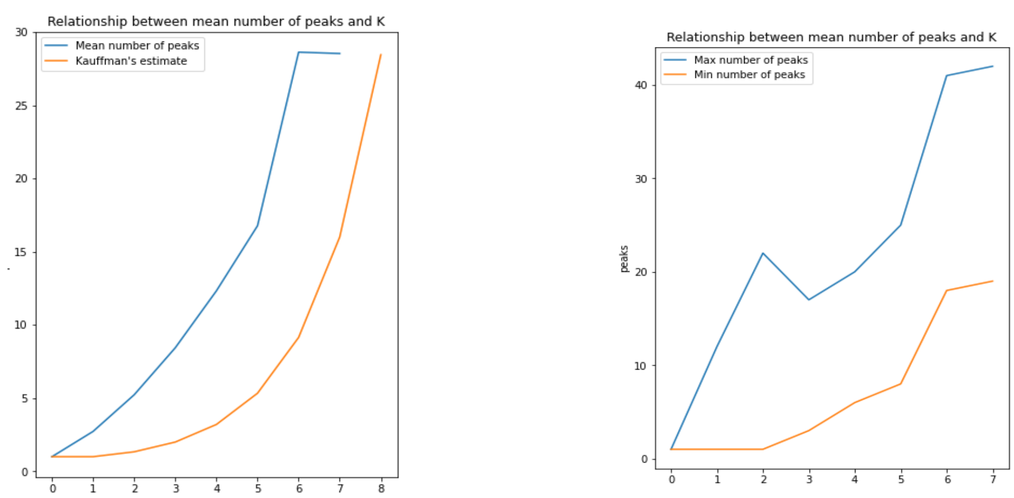

Figure 7 – mean number of local peaks through K

Mean value of peaks for N=K=8 is same as Kauffman's estimate of $\frac{2^N}{N+1} = \frac{256}{9} = 28.4$ It is clearly visible that Kauffman's estimate of number of local peaks is totally correct. Overall, model's behavior clearly shows that there is a relationship between increasing K and number of local peaks. Hence, we can make an assumption that increasing of K increases ruggedness of a generated random Fitness Landscape.

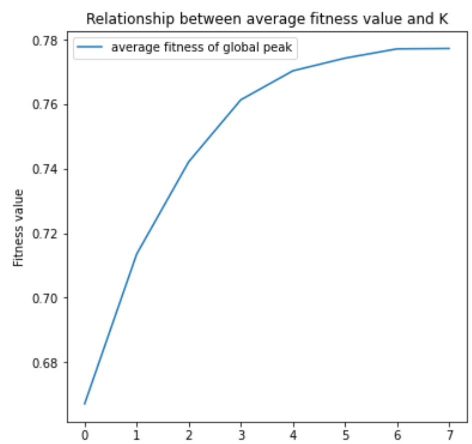

Figure 8 – average fitness of global peak number of local peaks through K

Finally, we can see that increasing K raises average fitness of global peak as well, which is one of many indications of importance of K in ruggedness of a fitness landscape.

3.2. Analyzing the impact on system's stability based on tuning K

This final part of the paper covers a peculiar question of whether or not changing two or more activation states makes impact on system's stability. We'll demonstrate the results on models with two changed activation states for $N = 6$ and $K = \{1; 2; 5\}$. Firstly, we'll uniformly generate an interaction matrix I of size NxN. After that, we'll diagonalize I and generate $N = 6$ fitness values in range $(0; 1)$ from normal distribution and call it A, a sequence of N states. A fitness function will be defined for the element i as follows:

$$F(i, K) = \frac{\sum_{i=1}^{N} I[i][i-K] * A[i]}{N}$$

Conclusion

This paper covers the basic principles of NK-modelling and the intuition behind tuning different values of K. In particularly, it describes:

NK-Boolean network models, ways of their representation

Concept of fitness landscape and a corresponding NK-model

Regimes of NK-models with regard to K and transition between them

Stability of NK-models with regard to K and changes to activation states

It was shown that ruggedness (smoothness) of the NK model, length of a trajectory, its stability and number of local peaks and its global fitness values are crucially determined by its external parameters N and K and the distribution of the generated landscape makes little to no impact on the aforementioned characteristics.

However, studying a model in an extensive way remains extremely computationally heavy, as the "catastrophe of complexity" for systems with K~N cumbers understanding the nature of both NK-model in particular and complex systems in general.

References

1. Kauffman S. The NK Model of Rugged Fitness Landscapes and Its Application to Maturation of the Immune Response / Kauffman S., Weinberger E. – University of Pennsylvania [1989] – 35c.

2. Kauffman S. A. – The origins of order. – New York Oxford: Oxford University Press [1993] – 731c.

3. Kauffman S.A. – At Home in the Universe: The Search for the Laws of Self-Organization and Complexity – New York Oxford: Oxford University Press [1996] – 336c.

4. Kauffman, S. A. – Metabolic stability and epigenesis in randomly constructed genetic nets – Journal of Theoretical Biology. 22 (3): 437–467. doi:10.1016/0022-5193(69)90015-0. – [1969]

5. Biggiero L. Relational Methodologies and Epistemology in Economics and Management Sciences – AFAE [2016] – 63c.

6. Somogyvari, Z., Payrits, S – Length of state cycles of random Boolean networks: An analytic study – Journal of Physics. A, Mathematical and General, 33, doi:10.1088/0305-4470/33/38/304 – Budapest [2000].

7. Weinberger, E. – Local properties of Kauffman's N-k model: A tunably rugged energy landscape. Physical Review A. 10. 44: 6399–6413. doi:10.1103/physreva.44.6399.

8. Derrida, B; Pomeau, Y – Random Networks of Automata: A Simple Annealed Approximation – Europhysics Letters (EPL). 1 (2): 45–49. doi:10.1209/0295-5075/1/2/001.- [1986]