ISTANBUL TECHNICAL UNIVERSITY ★ GRADUATE SCHOOL

SPARSE IDENTIFICATION OF NON-LINEAR DYNAMICS (SINDY) OF LANDSCAPE EVOLUTION MODEL SIMULATIONS

M.Sc. THESIS

Özgür Doğan BİROL

Department of Climate and Marine Sciences

Earth System Science Programme

JULY 2024

ISTANBUL TECHNICAL UNIVERSITY ★ GRADUATE SCHOOL

SPARSE IDENTIFICATION OF NON-LINEAR DYNAMICS (SINDY) OF LANDSCAPE EVOLUTION MODEL SIMULATIONS

M.Sc. THESIS

Özgür Doğan BİROL
(601211007)

Department of Climate and Marine Sciences

Earth System Science Programme

Thesis Advisor: Dr. Ömer YETEMEN

JULY 2024

İSTANBUL TEKNİK ÜNİVERSİTESİ ★ LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ

YERYÜZÜ EVRİM MODELİ BENZETİMLERİNİN DOĞRUSAL OLMAYAN DİNAMİKLERİNİN ARALIKLI OLARAK SAPTANMASI

YÜKSEK LİSANS TEZİ

Özgür Doğan BİROL
(601211007)

İklim ve Deniz Bilimleri Anabilim Dalı

Yer Sistem Bilimi Programı

Tez Danışmanı: Dr. Ömer YETEMEN

TEMMUZ 2024

Özgür Doğan Birol, a M.Sc. student of İTU Graduate School student ID 601211007 successfully defended the thesis/dissertation entitled “SPARSE IDENTIFICATION OF NON-LINEAR DYNAMICS (SINDY) OF LANDSCAPE EVOLUTION MODEL SIMULATIONS”, which he/she prepared after fulfilling the requirements specified in the associated legislations, before the jury whose signatures are below.

Thesis Advisor :	**Asst. Prof. Dr. Ömer YETEMEN** Istanbul Technical University	.....
Jury Members :	**Assoc. Prof. Dr. Yasemin EZBER** Istanbul Technical University	.....
	Assoc. Prof. Dr. Abdullah AKBAŞ Bursa Uludağ University	.....

Date of Submission : 24 May 2024
Date of Defense : 01 July 2024

FOREWORD

I would like to express my sincere gratitude to my advising professor, Dr. Ömer Yetemen. His stimulating conversations which enhanced my intuition and knowledge about landscape evolution modelling helped me overcome obstacles with ease. I am deeply thankful for his guidance and encouragement. It is also important for me to thank Bengisu Şensoy for she enlightened me about thesis submission process and overall school bureaucracy. Lastly, I am forever grateful to mother, father, and my brother for their always encouraging attitude towards me accomplishing a degree alongside my full time job.

July 2024

Özgür Doğan BİROL
İşletme Mühendisi

TABLE OF CONTENTS

Page

ABBREVIATIONS

LEM **:** Landscape Evolution Model

SINDy **:** Sparse Identification of Non-linear Dynamics

GP **:** Genetic Programming

LIST OF TABLES

Page

LIST OF FIGURES

Page

SPARSE IDENTIFICATION OF NON-LINEAR DYNAMICS (SINDY) OF LANDSCAPE EVOLUTION MODEL SIMULATIONS

SUMMARY

This study, titled "Sparse Identification of Nonlinear Dynamics (SINDy) of Landscape Evolution Model (LEM) Simulations" explores the linear inference of differential physical processes simulated by the LEM called Landlab by using the SINDy toolkit. The research aims to enable LEM researchers to incorporate data trained models to explain nonlinear phenomena alongside the traditional approach of physical equations.

The study employed an iterative approach to investigate explanation of the non-linear Taylor diffusion process from simulation data using a sparse, linear model. The data was collected from the simulations run by the LEM model Landlab which incorporated a non-linear hillslope and a fluvial flux process.

Results suggest that the underlying nonlinear dynamic could be identified with a trained SINDy model in a relatively sparse manner with a median average percentage error as low as 0.097%. Although, the simulations ran with the trained model exhibited drift from the original course of model data. This is expected due to stochasticity of non-linear models.

The findings of this study reveal that soil elevation at a node in a spatial grid can be inferred using the elevation values of the neighboring nodes as model features. This has important implications for LEM modelling providing viable employable alternatives to representative physical equations.

Additionally, this study contributes to the existing body of knowledge in earth sciences by showing that it is possible to linearize a process in spatial domain simulations when treated as a dynamical system.

Furthermore, the study highlights the challenge of reducing number of terms in a linear model to just a few, which can contribute to further research in this area.

In conclusion, this study sheds light on treating differential physical processes in a spatial domain as sparse linear models and providing valuable insights for researchers who design or employ LEM’s in their studies.

YERYÜZÜ EVRİM MODELİ BENZETİMLERİNİN DOĞRUSAL OLMAYAN DINAMİKLERININ ARALIKLI OLARAK SAPTANMASI

ÖZET

"Yeryüzü Evrim Modeli (YEM) Benzetimlerinin Doğrusal Olmayan Dinamiklerinin Aralıklı Olarak Saptanması" adlı yüksek lisans tezi Landlab adlı YEM'in benzetimlerindeki diferansiyel fiziksel süreçlerin doğrusal olarak açıklanmasını araştırmaktadır. Bunu kısaca SINDy adlı bir dizi metodu bünyesinde toplayan bir iskelet ile gerçekleştirmektedir. Bu çalışma YEM araştırmacılarına doğrusal olmayan görüngüleri açıklarken fiziksel formüllerin yanısıra veri ile eğitilmiş modelleri kullanabilme olanağını sağlamayı amaçlamaktadır.

Bu çalışma tekrarlı bir yaklaşım izleyerek doğrusal olmayan bir Taylor toprak yayınımı sürecini benzetim verisi ile aralıklı ve doğrusal olarak açıklayabilmeyi araştırdı. Veri ise Landlab adlı YEM modelinin doğrusal olmayan bir tepe eğrisi süreci ile flüviyal sürecinin dahil edilmesiyle alınan benzetim koşumlarından toplandı. Landlab Python programlama dilinde kullanılabilen modüler mantıkta kullanıcı dostu bir model kütüphanesidir.

Öncelikle 75 yatay, 75 dikey hücre tanımlanarak 2 boyutlu bir ızgara ile problem sahası tanımlandı. Izgara üzerindeki her bir düğüm arasındaki dikey uzaklıklar x ve y düzlemlerinde 100 metre olarak belirlendi. En fazla 90 metre olacak şekilde rassal yükselti farklılıkları ile başlangıç koşulları atandı. Izgaranın batı yönü toprak kütlesi giriş çıkışına açık bırakıldı. Benzetimin toprak kaybı yerine tepeleşme oluşturabilmesinin sebebi yılda 0.001 metre olacak biçimde yükselme değişkenidir.

Tepe eğrisi süreci diferansiyel bir doğada olan Taylor toprak yayınımı süreci olarak seçildi. Buna ek olarak toprak akısına suyun etkisini de dahil edildi. Her tekrar arasında 1,250 yıl olacak biçimde çalıştırıldı. YEM simülasyonunun her bir adımda gerçekleştirdiği şey basitçe komşu düğümler arasında tepeleşme ve flüviyal taşınmaları hesaplamak ve bunlara bir de yıllık yükselme miktarını eklemekti. Bu taşınmalar en dik iniş prensibiyle sağlandı. Tekrarlı biçimde değişkenleri iyileştirerek defalarca çalıştırdığımız ve 75 milyon yıl boyunca koşturulan benzetimin sonuçları dendritik bir tepe oluşumu çıktısı verdi. Benzetim koşulurken hücrelerin birleşim noktaları olan her düğüm ve onlara komşu 8 düğümün toprak yükseltisi bilgisi her zaman adımında toplandı.

SINDy araçlar bütünü çoğu fiziksel görüngünün bir kaç terim ile (yani aralıklı biçimde) açıklanabildiği varsayımına dayanmaktadır. Temel amacı yaklaşıklık üzerine kurulu bir biçimde doğrusal olmayan bir fiziksel görüngünün açıklayıcı formülüne en yakın çözüme yalnızca gözlem verisinden yola çıkarak ulaşmaktır. SINDy'yi önceleyen metotlarda aralıklı bir çözüme ulaşmak için geliştirilen yaklaşımlar çözüme ulaşmak için gereken işlemci süresi açısından çok ucuz değildi. Bu yeni metot aralıklılık şartını L1 düzenleyicisi ile sağlamakta. Bu düzenleyici, kullanılan dışbükey eniyileştirme algoritmasının aralıklı bir sonuca ulaşması için çözümde fazla sayıda terim bulunmasını cezalandırmaktadır. L1 düzenleyicisi kullanılan bir eniyileştirme

sürecinin en iyi çözüme yaklaşacağı varsayımı sıkıştırıcı algılama literatüründen gelmektedir.

Açıklayıcı birkaç terimin geniş bir aday terimler havuzundan geldiğini göz önüne aldığımızda SINDy tekrarlı bir biçimde bu çözüm uzayında en iyi çözümü arar. Aslında bu terimler doğrusal olmayan dinamik sistemin zaman serisi verisindeki aday açıklayıcı özelliklerdir. Örneğin, bir Lorenz sisteminde bu terimler x, y, z konumlarıdır.

Yalnız, bu terimlerin doğrusal olmayan biçimlerde de görüngüye etkidiğini düşündüğümüzde açıklayıcı formülün polinom veya trigonometrik gibi özellikleri aday çözüm terimleri havuzuna dahil olmalıdır. Üstelik terimlerin birbirleriyle olan ilişkileri de bu havuzda yer almalıdır. Bu çalışmada yalnızca polinom çözüm uzayı kullanarak varılan açıklayıcılık düzeyi yeterli bulunmuş ve bu tercih edilmiştir. Lorenz örneğinden hareketle çözüm uzayında örneğin x*y, x^2*y gibi terimler bulunacaktır. Bizim uygulamamızda ise bu aday terimler komşu düğümlerin kendileri ve birbirleri ile polinomyal kombinasyonlarıdır. Yani benzetimin üzerinde koşulduğu ızgaradaki bir düğümü çevreleyen 9 komşusu aday çözüm havuzunu oluşturan özelliklerdir.

SINDy doğrusal olmayan bir görüngüyü bu şekilde doğrusal bir probleme dönüştürür ve yaklaşık olarak açıklar. Bu doğrusal model eğitimi sürecinde STLSQ adlı eniyileme algoritmasını kullanır. Bu algoritma en aralıklı çözümü bulmak için aday çözüm uzayındaki terimlerle modeller eğitir. Eğitim sürecinde modele dahil edilen özellik adedini bir ceza etkeni olarak tutarak yalnızca en ilgili terimlerin kalmasını sağlar. Eğitimden sonra ise katsayısı çok düşük olan terimleri modelden çıkararak başa döner. Bu iyileştirme yaklaşımı sonunda çözüm uzayından doğrusal olmayan dinamiği en çok açıklayan terimlerin seçilmesini sağlar.

STLSQ algoritmasının düşük katsayıya sahip terimleri çözümden çıkarmasını belirleyen eşik değer değişkeni çalışmanın temel eniyileme çabasını oluşturdu. Öncelikle bu değeri 0.1 ile 5 arasına 20 adet değer düşecek şekilde serbestçe belirleyip modelleri eğittiğimizde verimli sonuçlar vermeye devam ettiğini gördük. Aralığı genişletip 150'ye kadar değerlerle eğitim yapmayı denedik.

Tüm eğitim süreçlerinden sonra hangi modelin en iyi sonuçları verdiğini araştırdık. Öncelikle bunun için YEM benzetimindeki her bir zaman adımındaki durum değişkenlerini, yani komşu düğümlerin toprak yüksekliklerini modellerin çıktıladığı çözümlere girdi olarak besleyerek tahminleme yaptık ve hatalarını ölçtük. Burada tespit ettiğimiz hatalar ihmal edilecek derecede küçüktü.

İkincil olarak tüm modelleri YEM benzetimindeki başlangıç koşullarıyla 200. zaman adımına kadar her bir adımdaki çıktıyı bir sonraki adımı tahminlemeye girdi olarak besleyerek tahminledik. Bunun sonunda elde ettiğimiz model çıktılarını 3 farklı gösterge kullanarak hata hesapladık.

Ortalama mutlak yüzde hatası dediğimiz hata oranı YEM benzetimiyle model çıktılarının birebir karşılaştırılmasıyla elde ettiğimiz hataların ortalamasına işaret ediyor. Her ne kadar benzetim çıktılarının doğrusal olmayan doğası sebebiyle asıl veriden farklı olacağı, yani zaman adımları boyunca farklı bir rotaya sürükleneceği bilinse de bu sürüklenmenin ne derece farklı olduğu model seçiminde bir faktör olarak kullanılabilirdi.

Kullandığımız diğer iki hata hesaplama göstergesi asıl veri, yani YEM çıktılarının dağılımına ait özelliklerin model tahminlerinin dağılım özellikleri ile karşılaştırılması ile ilgili idi. Burada dağılımların ortalamalarını ve standart sapmalarını karşılaştırdık.

Asıl veriden farklı olmama düzeyi aslında istatistiki anlamda aynı anakütleden gelebileceklerine dair destekleyici bir bulgu olacaktı.

Bu üç farklı hata türünün ağırlıklı ortalamasını alarak belirleyici bir model seçimi hata skoruna erişmeyi hedefledik. Burada isteğimize göre belirlediğimiz katsayılar hata türlerinden bahsettiğimiz sırayla 0.3, 0.5, 0.2 oldu. Buna göre seçtiğimiz model aralıklılık katsayısı olarak 125 ile eğittiğimiz oldu.

Sonuçlara göre sistemin altında yatan doğrusal olmayan dinamikler bir SINDy modelinin eğitilmesi ile görece aralıklı olarak ve %0.097'ye kadar düşük bir hata oranıyla açıklanabiliyor. Yine de vurgulamak gerekir ki eğitilen model ile alınan benzetim koşumları asıl model verisine göre bir miktar sapma gösterdi. Bu doğrusal olmayan modellerin doğasındaki rassallık ile alakalıdır ve beklenen bir sonuçtur.

Bu çalışmanın bulguları gösteriyor ki benzetimdeki uzamsal bir ızgara kurulumunda bir düğüm noktasının toprak yüksekliğini komşu düğüm noktalarının yüksekliklerini özellikler olarak kullanan bir istatistiksel öğrenme modeli ile açıklamak mümkündür. Bu durum YEM modellemede bu metodun fiziksel denklemlere alternatif olarak kullanılabilmesi konusunda önemli bir sonuç getirmektedir. Yine de bu metodun teorik bilimsel açıklamaya bir alternatif değil, yardımcı olan bir metot olduğunu açıkça vurgulamak gereklidir.

SINDy ile toprak akısı süreçlerini yaklaşık olarak açıklamakta bizim kullandığımız yöntem komşu düğümlerin yükseklik verilerini kullanarak altta yatan fiziksel bilgiyi dolaylı yoldan öğrenmek oldu. Bu durum SINDy'ye yerellik sonucunu getiriyor. Yani bizim izlediğimiz prosedür oluşturduğumuz ızgara ve üzerinde koştuğumuz benzetimin spesifik şartlarını öğreniyor. Tepe eğrisi veya flüviyal süreçleri genelleştirilebilir bir biçimde öğrenmek için dolaylı bilgi veren değişkenler yerine fiziksel değişkenlerin kullanılması gerekir. Yine de bu yerellik durumu bu metodun fiziksel değişkenleri gözlemlemenin limitli olduğu şartlarda kullanılabilir, verimli bir metot olarak öne çıkarıyor.

Buna ek olarak, zaman kaynağının dar olduğu ve bilimsel açıklama süreçleri için yeterli kaynağın bulunmadığı araştırma kurgularında, doğrusal olmayan fiziksel süreçleri yaklaşık bir biçimde öğrenmek SINDy ile mümkündür.

Bazı araştırmalarda gereken yalnızca sistemi etkileyen bileşenlerin bir kısmının tahmin edilebilir derecede açıklanması olabilir. Yani amaç bu bilinmeyeni tahmin edilebilir kılıp ana araştırma konusuna yardımcı olarak etkinleştirmek olabilir. Bu durumda SINDy mevcut olan gözlem verisi üzerinden sistemin gereken bileşenini açıklayarak ana araştırma amacına varılmada kullanılmasını sağlayabilir.

Bu tez dünya bilimleri çalışmalarındaki bilgi birikimine uzamsal bir ızgara bazlı benzetim koşumlarının süreçlerine dinamik bir sistem gibi yaklaşıldığında onları doğrusal bir probleme dönüştürmenin mümkün olduğunu göstererek katkıda bulunuyor.

Dahası, bu çalışma doğrusal bir modelin terim sayısını azaltarak bir kaç adede düşürme çabasının zorlayıcı doğasını vurgulayarak bu bilginin ileride yapılacak ilgili araştırmalara katkıda bulunmasını mümkün kılıyor.

Sonuç olarak bu yüksek lisans tezi uzamsal bir düzlemde gerçekleşen diferansiyel fiziksel süreçlere aralıklı lineer modeller olarak yaklaşmaya ışık tutuyor. YEM tasarlayan veya kullanan araştırmacılara değerli içgörüler sağlıyor.

1. INTRODUCTION

1.1 Background and Context

1.1.1 General introduction

Most engineering research fields including earth science are interested in understanding the natural phenomena that manifest as complex nonlinear dynamics. Traditionally, researchers have utilized mathematical equations to numerically or approximately infer these dynamics. The methodology imvolves resolving the physical laws governing processes such as the conversation of mass. This undertaking requires the knowledge of the underlying physical processes and necessitates a delicate balance between complexity and robustness (Prada et. al, 2018).

The second approach is to bring these equations to light using only the observation data. The approach here is to make an attempt to uncover an underlying model by using data obtained from certain experiments. This model is obtained by putting an algorithm to use. This algorithm is expected to yield the parameters of the model. This means that the candidate model type, the algorithm, and the model discovery process design are the questions which the researcher is expected to enlighten.

Although the confidence in data driven models could be considered high due to observations being the only source of truth, they are challenging in terms of generalization. Unlike the physical equations, the data of the experiment can be domain specific. As a result, this might cause the obtained model to apply only under certain conditions (Prada et al., 2018).

There have been several methods employed to discover nonlinear systems since 1980's. There are a few outstanding approaches: Delay coordinate embedding method, mutual information method, differential equation fitting, and a more recent practice of compressed sensing (Wang et al., 2016). According to this latter approach, most natural phenomena could be inferred by functions which are smooth to the level that will enable them to be approximated by finite series expansions.

The contrast between the data-driven or physical equations has been interesting to address. Both approaches were criticized from different perspectives. Data-driven models were claimed to find correlations between data without addressing the natural meaning behind their subjects while physical equations require a lot of accumulated scientific knowledge to hypothesize the relationships between the components of the system (Prada et al., 2018).

Various symbolic regression methods were devised to overcome this dichotomy. It was proclaimed to provide scientific community with a method to equip a purely data-driven method that searches for the most meaningful equations alongside their parameters (Schmidt and Lipson, 2009). However, the symbolic regression is also criticized for being expensive in terms of processing resources (Brunton et al., 2016).

As a response, Brunton et al. (2016) devised Sparse Identification of Nonlinear Dynamics (SINDy) by combining early approaches of sparse regression, and compressed sensing. The essential motivation behind this method is to make use of the case that most of the time only a handful of features govern the dynamics of natural systems. The model aims to establish a fine balance between complexity and generalizability.

1.1.2 Research problem

SINDy toolbox was shown effective in identifying non-linear dynamics of a system before in numerous fields of research. The fundamental purpose of this study is to investigate whether the SINDY method can robustly and sparsely discover a non-linear soil diffusion process within a spatial domain. We seek to demonstrate that earth science researchers can employ similar practices to uncover governing equations within spatial domains, even without the prior knowledge of the physical equations.

1.2 Research Objectives

Although SINDy method has been applied to numerous fields such as fluid dynamics, turbulence, optics, computational chemistry, we could not identify a study that investigated its applicability to a non-linear soil diffusion process in Landscape Evolution Model (LEM) simulations. We seek to incorporate the SINDy method as a suitable system identifier for our problem setting in the Landlab simulations which is a modular and grid-based LEM provided in Python language (Hobley et al., 2017). We

perform the simulation experiment for a specific setting leaving the generalization problem to further research.

1.2.1 Main research objective

Our primary mission is to develop a simulation of soil diffusion on a spatial domain, of which dynamics will be interpreted by SINDy. We intend to define the domain as a traditional grid system which will facilitate processes of fluvial transport and soil diffusion. The system will calculate the soil elevation values for each node of the grid.

The simulation data will be delivered to SINDy for interpretation. The model features will consist of the soil elevation states for each 9 nodes forming a 3X3 matrix. This means that the model does not intend to capture the equations of the physical processes. Instead, it resolves how the neighboring nodes exchange sediment using soil elevation due to the conservation of mass principle in the underlying physical equations.

The discovery of this dynamic will show that the linearization of a differential process with interdependent spatial features is possible without further input features. In our experiment, the simulation data is noise-free. It is left for further research to address if added noise or direct field observations could achieve similar results successfully.

Our purpose will be achieved at the end where we conceptually emphasize that the relationship between the nodes could be discovered to exhibit the behavior of the system under stable conditions.

1.2.2 Subsidiary objectives

Aside from our main purpose, we aim to identify the optimal level of sparsity where the model prediction error remains within acceptable levels, and the equations contain the fewest possible terms. This is important to address to find easily comprehensible equations. The SINDy model provides a direct way to interact with sparsity of the model using the “threshold” parameter. We seek to perform an iterative grid search to find the most desired value. Having a single hyperparameter to optimize for, our parameter grid space will be one dimensional.

The SINDy library offers different feature libraries similar to symbolic regression for uncovering the equations. The key difference is that the researcher decides which

feature library or combinations of libraries to use. We employ polynomial library of second degree in this experiment. This means that the input features are exponentiated up to second order, and their combinations are treated as features of the sparse regression model. As a side objective, we will have testified that the polynomial library is fit for discovering dynamics in a spatial domain of an LEM.

1.3 Literature Review

Traditionally, the researchers preferred two different methods to infer the nonlinear dynamics in physical processes. Firstly, the physical equations that reconstruct the physical processes were developed through series of experimentation and validation of the findings with observation data. In contrast, data-driven methods initially disregarded the relationships between the system components. They had been criticized to train data to discover pure correlations without inference before they began to incorporate a priori relationships into their dynamic discovery. The initial studies which involved a priori relationships were published by Rao (1965) and Bard (1974). Upon that, the studies which utilized similar methods expanded in various field applications (Cozad et al., 2014).

One of the most significant branch of methods that combined the a priori knowledge of a system to the process of discovery of the dynamics of it started with genetic algorithms (Holland, 1992). It was inspired by Darwin's survival of the fittest schema which described the progressive genetic heritage of the species with survival and breeding. The mixing of the genes through breeding, the random mutations, and survival due to stronger properties determines the genetic composition of the next generations to come.

The genetic algorithm is based on searching for optimal place in the solution space by simple binary inclusion or exclusion of certain features. Starting with random set of candidate solutions, each iteration cross-breeding (as shown in Figure 1.1), random mutations make up the next generation of candidate solutions. The fitness score (such as the magnitude of error) determines if the candidate solution suffices. Then those of high quality breed and the other ones are eliminated. Generation by generation, the solutions progressively improve (Holland, 1992).

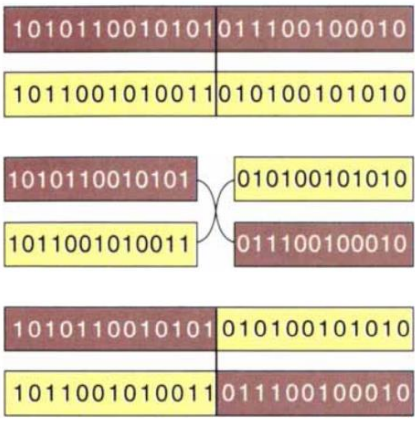


Figure 1.1 : Cross-over of portions of candidate solutions (Holland, 1992).

A study which attempted to discover nonlinear system dynamics of twin-tank hydraulic system by first clear-box model (pure physical equations that are known), secondly the black-box model (a gradient optimization algorithm), and lastly with a grey-box method tested the robustness of genetic algorithm. It equipped a variation of a genetic algorithm with specific set of mutation rules and performed further optimizations with a gradient based algorithm. The result was that the grey-box algorithm based on genetic algorithm was the most successful among the three methods (Tan & Li, 2002).

The next significant development was the derivation of genetic programming (GP) as a subset method of genetic algorithm. (Koza, 1992). Unlike the binary selection of genetic algorithm, the GP constructs syntax trees for candidate solutions as shown in Figure 1.2. The output of GP is a function. Rather than inclusion or exclusion of parameters, it is founded on selection of functions and parameters for branches. Reproduction, crossover, and mutations are executed as genetic operations during each iteration. Crossover is performed by selection of random nodes and exchanging the branches of the syntax trees (Koza & Poli, 2006).

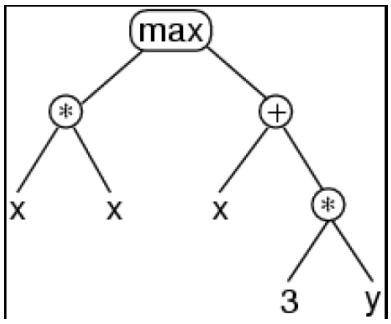


Figure 1.2 : GP output function represented as tree; max(x*x,x+3*y).

Plenty of improvements and variations to genetic programming have been published and used to identify non-linear systems under the umbrella term symbolic regression since then. Bongard et al. (2006) have applied a special enhancement to genetic programming based symbolic regression by adding principles of partitioning, probing, and snipping for the objective of identification of nonlinear systems in a significant study. Partitioning attempts to explain each variable by their own functions to reduce the size of the number of candidate solutions. Probing transforms the testing process into an active learning scheme by picking different initial states for the system in question and see how the model in question will perform against different trajectories of the system. Snipping, an optimization principle offers two benefits: preventing overfitting and simplifying models. It is based on replacing the sub-expressions of the model in question with a constant.

Another genetic programming inspired method was called the linear symbolic regression which had a narrower focus on identification of the interpretable and concise mathematical formulas. Initially, the space of primitive functions generated by nonlinear combinations (including arithmetic operations, mathematical functions, conditional logic operators, or domain specific functions) of system features is performed. The purpose is simply to transform a non-linear problem into linear inference. Then the problem is solved with a convex optimization (in other words, the search for the best coefficients) by fitting an optimization algorithm to the observations (Makke & Chawla, 2024).

Although the genetic programming based symbolic regression methods are effective, they do not scale well. As a result, it can be computationally expensive. This is because they do a brute-force scan of the solution space. Meanwhile, linear symbolic regression methods can utilize optimization methods to approximately converge to a robust fit. Despite that, most of those algorithms converged to solutions that are not sparse enough. In simpler terms, the output formula tends to have many coefficients that are not 0.

This sparsity condition is needed for data-driven identification of non-linear dynamics systems which are physical systems in nature. Most equations of the physical systems are widely known to consist of a few terms (Brunton et al., 2016). According to the contemporary research concerning compressed sensing, it is a high probability result that convex optimization with sparsity regularization will converge to a sparse solution with compressive sampling (Barainuk, 2007).

As a response, Brunton et al. (2016) developed the SINDy approach which is an approximation method for uncovering of the nonlinear dynamics. All that SINDy shares in common with the genetic programming is limited to generation of a basis function library. Similar to linear symbolic regresison, this library consists of a large set of terms from the possible solution space.

SINDy is based on the assumption that the most physical systems are explained by functions with a few terms. This assumption turns the problem into a sparse optimization objective with only a few members of the space of the basis functions will be included in the output function. Unlike genetic algorithms which attempt to converge to the optimum solution by random mutations in a relatively indeterministic way, SINDy attempts to tackle the model discovery by optimizing a linear regression by a special sparsity algorithm called sequentially thresholded least squares algorithm (STLSQ).

Equation (1.1) shows the approximation problem of SINDy where X is the matrix of the observed values of the dynamical system, while $\dot{X}$ is the matrix of the differentiated values. They are the x_{t+1} values in a system with discrete time. The goal is to find a sparse vector Ξ which hosts coefficients for only a few of the whole set of basis functions $\theta(X)$.

$$\dot{X} \approx \theta(X)\Xi \tag{1.1}$$

Equation (1.2) describes the objective function of STLSQ optimization.

$$\min\left\|\dot{X} - \theta(X)\Xi\right\| + \lambda\|\Xi\|_0 \tag{1.2}$$

The first term of the equation represents the error which is the main criterion for the fit. The second term is based on setting a threshold constant, λ which penalizes having more members in the vector for it decreases the sparsity. The value of λ controls the number of basis functions with non-zero coefficients. Although, Tibshirani (1996) suggested L1 Lasso regression to minimize the coefficients of the fit, SINDy is completely interested in the number of coefficients involved in respect with sparsity assumption. This is because of the previously explained theoretical findings that support converging to a sparse solution with this type of regularization.

While SINDy penalizes the number of basis functions in the fit, it also offers an iterative approach to find the sparsest solution possible by eliminating terms with small coefficients after each fit. When the small coefficients are eliminated, the remaining basis functions are refit. As a result, the output function is expected to have coefficients that are small. This process continues until there are no marginally small coefficients. Figure 1.3 schematically summarizes how SINDy processes Lorenz system data and predicts the underlying equations for 3D coordinate system.

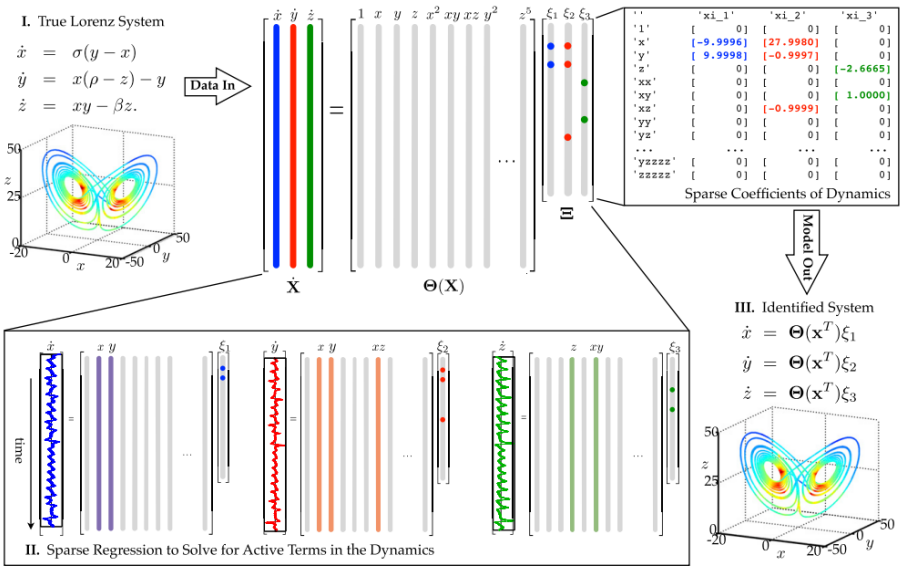


Figure 1.3 : SINDy applied on Lorenz System (Brunton et al., 2016).

2. METHODOLOGY

The purpose of the study is to infer a nonlinear dynamical system from time series data in the broadest sense. More specifically, we are interested in explaining the dynamics from the data generated through a LEM that incorporated non-linear processes for a specific spatial domain. The LEM modelling tool we employ for this study is Landlab, a modular software that enables researchers add/remove components needed for their specific problem. The data generated by model run is used as input for SINDy to infer dynamics on a granular piece of the spatial domain. The resulting model is generalizable for any sub-region, and any initial time t_0 across the domain.

2.1 Data Generation Through Landlab Simulation

Landlab model is utilized to create a 75×75 grid system with $d_x = d_y = 100$ meters. The boundaries at the left edge were set open, meanwhile the rest of the edges were closed for material exchange. The initial height, z_0 of all nodes were set to 1. Then added with random integer noise varying between 0 and 90 with the NumPy random seed set to 0. We determined this number arbitrarily, but large enough to facilitate horizontal mass movement for fluvial processes as they use the steepest slope to determine flow direction. Figure 2.1 illustrates the created grid below.

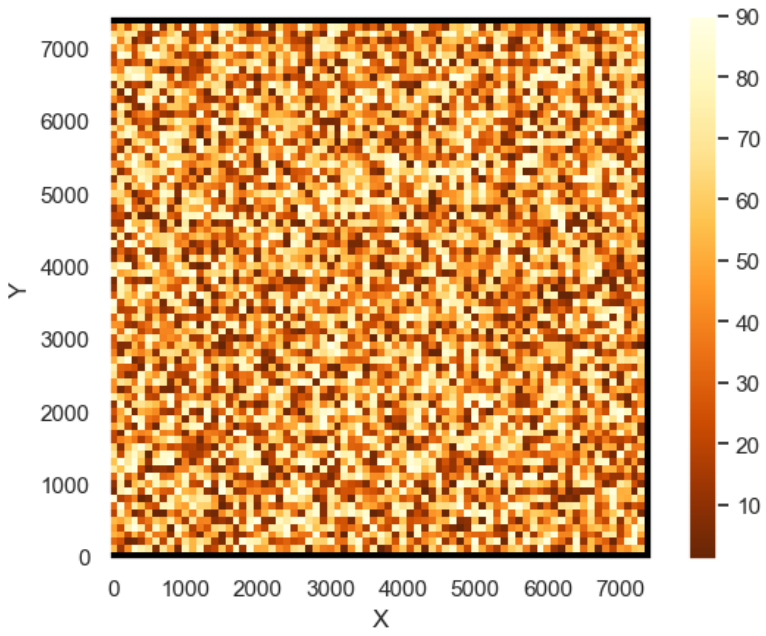


Figure 2.1 : The initial grid with node heights (m).

We ran hillslope evolution process by utilizing the Landlab component TaylorNonLinearDiffuser which evaluates the nonlinear hillslope soil flux by the method described in Ganti et al. (2012). It applies a Taylor series expansion to the variation derived by Andrews-Bucknam. This process is what brings the non-linearity to the model. Equation (2.1) shows this expansion below (Barnhart et al., 2019).

$$q_s = K * S * (1 + \left(\frac{S}{S_c}\right)^2 + \left(\frac{S}{S_c}\right)^4 + \cdots + \left(\frac{S}{S_c}\right)^{2(n-1)} \quad (2.1)$$

According to above formula K is diffusivity constant, S corresponds to slope, and S_c is the critical slope. As the slope increases, the non-linear portion of the equation receives more weight, thus exponentially increasing the amount of soil flux. This component gives the non-linear nature to the system. Hence it makes the simulation a good candidate system for SINDy to resolve.

The second component used in the LEM simulation is called StreamPowerEroder and it represents the fluvial part of the total soil transport. This component is used in accordance with the component FlowAccumulator to route the flux calculated by it. The flow direction logic used is the steepest slope.

The model parameters and constants were set in parallel with the official examples. For instance, the uplift rate (m/yr) was set to 0.001 which is enough amount to generate enough soil lift in a short timeframe to compensate for the hillslope and fluvial soil flux processes (see Appendix A for the constants and values). The only variable we have modified was the hillslope transport coefficient K_{hs}. We have decreased it enough to allow for a constant timestep.

The timestep (d_t) was set according to the Courant stability condition, as shown in Equation (2.2), where K_{hs} refers to hillslope transport coefficient. Correlated with distances between the nodes (d_x), the timestep needed to be small enough to provide stability of the runs of the hillslope process due to the approximating nature of Taylor series expansion. After setting up the parameters and constants, the Landlab modules were initiated.

$$dt = dx * dx/(7 * K_{hs}) \quad (2.2)$$

The data which we intended to generate for model training consisted of elevation (z) values of each 9 neighboring nodes on the grid at any timestep of the simulation.

The simulation was run with 60,000 iterations which is an arbitrary number considered enough to generate amount of data to train SINDy. Considering that a timestep is 1,250 years, the total simulated time amounted to 75,000,000 years. The run and data collection took almost an hour to complete in a personal computer with an Intel Core i5 processor. Figure 2.2 shows the distribution of height on the grid after the simulation run. The fluvial channels are present in this representation.

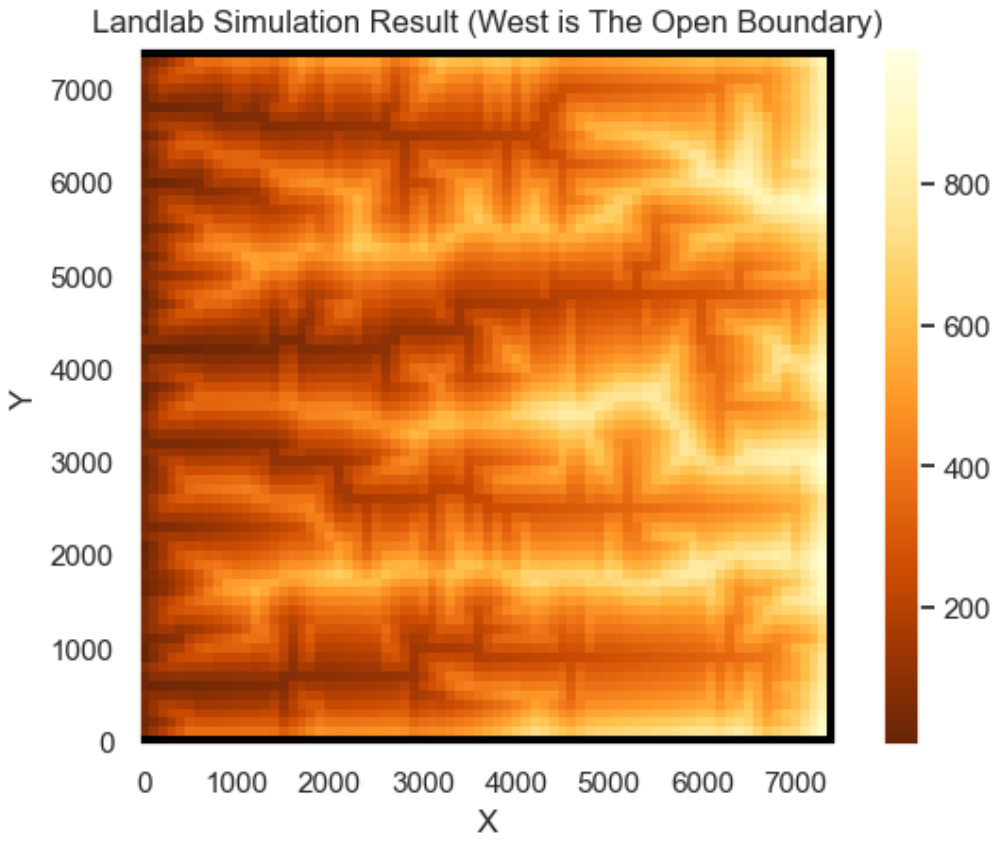


Figure 2.2 : The grid after simulation run (m).

When we take a look at the 3D representation on the Figure 2.3, the average slope and the result of an accented fluvial component become more apparent.

The hillslope formation is best represented by average elevations of all 75 vertical columns, and taking their moving averages one more time to smooth the data in Figure 2.4, below.

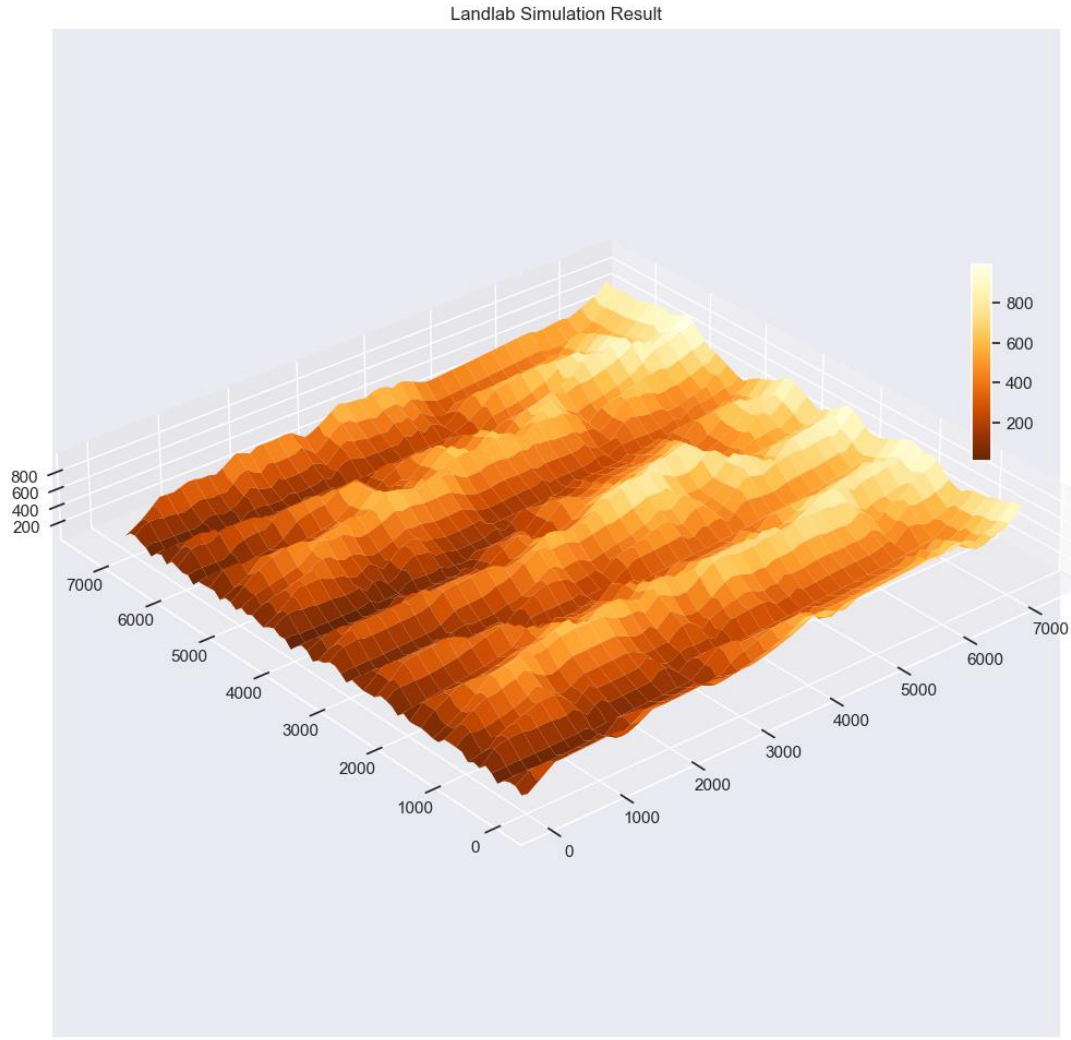


Figure 2.3 : The 3D representation of the simulated hillslope (m).

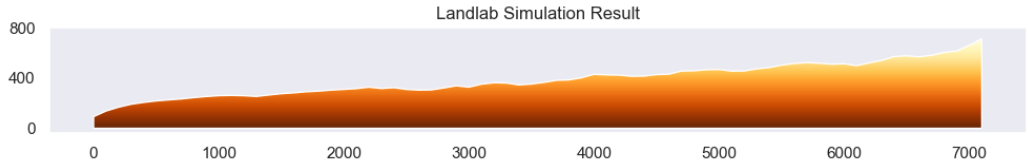


Figure 2.4 : The smoothed slope of the simulation (m).

2.2 Model Training Procedure by SINDy

The data generated for model training consisted of the time series information of each neighboring set of 9 nodes on the spatial grid. This way, there are many different time series trajectories to infer the dynamics of the soil flux in the designed simulation. Figure 2.5 shows the nature of different set of nodes on the spatial domain.

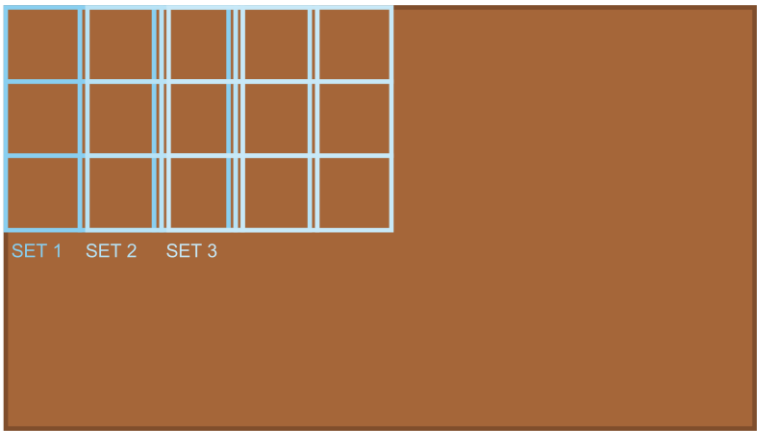


Figure 2.5 : The sets of 9 neighboring nodes.

After we obtain the data, we split data into train and test sets. Train part of the data was used for SINDy model training. We allocated 25% of the simulation runtime to training, and the rest for the testing. We ran 60,000 iterations which yielded enough amount of data to train by obtaining only a quarter of it.

Declaration and setting the hyperparameters of SINDy was performed (see Appendix A). SINDy uses finite differences method among other selectable options to differentiate the data. We did not utilize this in our training since all the timesteps (dt) were equal. Instead, we used the value of the next timestep (x_{t+1}) to indicate dx_t.

There are a lot of feature libraries available as candidate functions such as polynomial, trigonometric, or Fourier in pySINDy. The researcher is expected to make their feature library selection accordingly with their assumptions about the nature of the underlying physical phenomena. We decided to go with the polynomial library to start with and test different libraries if we fail to obtain a fit as we already know the nature of the hillslope and fluvial processes. They fit into the polynomial functions.

As the optimizer algorithm, we pick STLSQ which is unique to SINDy which finds the best sparse fit as explained. The critical decision is to set λ to a value which will minimize the number of features in the model while trying to optimize for error. There is no educated approach to find the best λ except to perform a brute-force search that locates a sweet spot between the model complexity and the error.

Initially, we trained SINDy for λ values ranging frım 0.1 to 50 with a step size of 5. This practice created 9 threshold values to train for. However, we realized that the models were far from sparse. Consequently, we expanded our training including λ values up to 150 with a step size 5.

2.3 Model Performance Evaluation

Our purpose is to solve for nonlinear dynamics rather than predict for numerical accuracy. In this sense, the most intuitive evaluation of model performance is estimation of average error for what we trained the models for. Equation (2.3) illustrates this calculation of mean error where $\hat{y}(t)$ is the model function which predicts the values across all the timesteps (n) whereas y_{t+1} is the original LEM output.

$$ME = \sum_{t=0}^{n} (\hat{y}(t) - y_{t+1})/n \tag{2.3}$$

When we applied this, all the models performed almost equally well regardless of their threshold values. However, as the dynamic under observation is a timeseries, it is also worth simulating the timeseries with the trained model starting from a certain point t and continue. Each model is likely to drift from original course and end up creating their own course within model constraints.

Although our purpose is rather seizing the essence of the dynamic rather than to mimic the identical course of values, the amount of drift could be useful in selection of the best model. In this light, we decided to use below 3 metrics to decide the performances of the models:

i) Model simulation mean error: Even though the models drift through time, magnitudes of the error of mean of their values could be assumed to stay within certain range if the trained model is able to understand the underlying dynamics.

ii) Model simulation standard deviation error: The same applies for the standard deviation error of the model simulation values compared to the original course of the simulation.

iii) Mean absolute percentage error of values: Comparison of model simulation results with the LEM simulation and observing the amount of drift from the original course.

We simulated the course of the timeseries with the trained models for the purpose of evaluation of their performances. We stopped the simulation at the 200th timestep,

arbitrarily. Below Table 2.1 shows the calculated errors of the models with different sparsity threshold values.

Table 2.1 : The errors of trained SINDy models.

Model Threshold (λ)	Mean Error	Std.Dev. Error	MAPE Error
125	2.14	1.946	0.0601
135	2.32	2.008	0.0617
140	2.28	2.021	0.0621
15	2.34	1.271	0.0695
20	2.58	0.472	0.0712
25	2.60	0.307	0.0714
110	2.49	1.717	0.0715
85	2.32	2.070	0.0729
130	2.41	1.633	0.0729
30	2.48	0.286	0.0754
80	2.31	1.657	0.0772
100	2.42	1.836	0.0803
145	2.48	0.519	0.0810
0.1	2.40	1.395	0.0815
55	2.11	0.380	0.0899
50	2.19	0.757	0.1105
35	2.41	7.537	0.1112
10	1.40	5.127	0.1376
60	2.42	6.961	0.1410
5	91.35	10803.598	66.9926
40	2.54	2.191	67.0696
70	3.17	33.553	49392.8791
105	3.13	4.899	51477.8124
120	2.14	0.073	52657.4687
115	1512785.65	293017003.766	19151312
65	3.31	6.096	39738466
75	3.41	23.227	655525927085e8

Seemingly, the 3 different types of errors we decided to evaluate do not have a correlation with each other. At this point, we needed to prioritize the error types and use a weight approach to come to a final decision. We have ranked the models for how they scored on 3 types of error 1st rank pointing to the lowest error. We consider that as much as the estimator of drift from original courses (MAPE error) gives us information about the model performance, we acknowledge that the numerical accuracy is the least priority against the statistical representation of LEM simulation population in the results.

This means that we decided the highest priority error metric is the mean error. Difference of standard deviation error yields relevant statistical information. Because we already assign the highest weight to mean error, standard deviation error could be considered the lowest weight error type. This leaves the MAPE error in the middle.

Hence, we decide to assign weights 0.5, 0.3, and 0.2 to mean, MAPE, and standard deviation errors, respectively. According to this, the best model is the model having the threshold value of 125. Table 2.2 shows the weighted errors.

Table 2.2 : The rank based errors sorted by weighted cumulative error score.

Model Threshold (λ)	Mean Error	Std. Dev. Error	MAPE	Weighted Score
125	3	14	1	4.6
55	2	4	15	6.3
140	6	16	3	7.1
15	10	8	4	7.8
135	9	15	2	8.1
50	5	7	16	8.7
80	7	11	11	9
120	4	1	24	9.4
85	8	17	8	9.8
10	1	20	18	9.9
130	12	10	9	10.7
0.1	11	9	14	11.5
30	17	2	10	11.9
20	20	5	5	12.5
25	21	3	6	12.9
145	16	6	13	13.1
100	14	13	12	13.2
110	18	12	7	13.5
35	13	23	17	16.2
60	15	22	19	17.6
40	19	18	21	19.4
105	22	19	23	21.7
70	23	25	22	23.1
65	24	21	26	24
5	26	26	20	24.2
75	25	24	27	25.4
115	27	27	25	26.4

At this point, it would be beneficial to simulate the complete timesteps to compare the predictions of the SINDy model with the Landlab simulation on the landscape grid plot. The stability problem prohibited us from doing that. Unfortunately, the python library of SINDy has stability problems.

Finally, we calculated the original error metric with the selected model with $\lambda = 125$ as we have shown in Equation 5. The MAPE error of model predictions from one timestep to the other scored 0,09%.

3. CONCLUSIONS

3.1 Summary of the Key Findings

This study was performed as an attempt to uncover the non-linear nature of soil elevations on a grid with an approach of data driven inference. Initially, we defined a grid with arbitrary soil elevations. We ran a simulation on the LEM model software called Landlab incorporating soil flux processes with a run time about 75 million years each timestep spanning 1,250 years. The model run incorporated a static uplift rate, a linear fluvial soil flux process, and a differential hillslope process. We have collected the soil elevation data throughout the simulation runs.

We have used the simulation data as input for SINDy toolkit which at first generated the 2nd order polynomial combinations of the existing features. These are employed as candidate terms of the model which explains the underlying dynamic. Then, SINDy fit linear models with the objective of adding the least number of terms to the model. This is controlled by the sparsity parameter lambda which iteratively reduces the number of features in the candidate solution by removing the terms with small coefficients.

Finally, we evaluated the trained model performance by predicting unseen portion of data. We repeated this process many times to find the best lambda parameter that yielded the sparsest model with a tolerable error.

We have found that it is possible to approximately infer the soil flux dynamic on a spatial grid domain by making use of this procedure of ours with acceptable amount of error (MAPE 0.09%.). It is worthy of notice that choosing the polynomial library for generation of the feature space provided enough basis of interpretation.

Our experiment did not use the physical features such as slope which almost always takes place in physical equations of soil flux processes. Instead, we sought the affirmation of the capability of SINDy through the soil elevation time series data of neighboring nodes only. We found that SINDy could thrivingly use the soil elevation of neighboring nodes as proxy information without requiring the purely physical variables.

This study contributed to the overall understanding of the soil flux dynamics by showing that a sparse linear algorithm with candidate functions could learn the essence

of the dynamics under the lens even when it is inherently non-linear. We have demonstrated that we could approximate these dynamics using polynomial function combinations generated from neighboring nodes on the spatial grid.

In conclusion, these results shed light on the possibility of utilizing trained models to compensate for lack of physical equations in an earth science phenomenon. This data driven procedure may be highly convenient for certain use cases. It traditionally requires a certain timeframe for researchers to explain a process scientifically through equations. There might be studies of which main purpose is not to explain the driving mechanism of a physical process. They could rather be concerned with the ability to simulate the process for understanding its interaction with other components in the system. Additionally, even when the purpose is to explain the physical process scientifically, the ability to approximate the process only from data with a set of candidate terms can shorten the duration of the modus operandi of the scientific inference process. Finally, it can be a productive undertaking to use this procedure for problem domains where there are physical (for instance climate conditions), financial (such as expensive equipment), or human limits to field data collection for physical interpretation but enough means to collect data for a proxy approximation approach like ours.

3.2 Limitations and Further Research

It is also important to acknowledge the limitations of this study. Firstly, the model training process is highly sensitive to locality. As SINDy searches for the best sparse combination of candidate function terms, it uses the collected observation data. If there are environment factors which affect the course or the coefficients of the non-linear dynamic, data will implicitly represent with this information. As the training process fits to the data, it inevitably inherits the local factors. This means, if the dynamic reacts differently in other domains due to environment variables, the resulting function from SINDy has to train with data from this specific environment. For example, if our experiment took place in a domain where the uplift rate or the hillslope rate coefficient were in different magnitudes, we would not be able to incorporate these changes into the resulting function. This is due to our output function being based on the soil elevation amounts of neighboring nodes. As a result, it becomes challenging to generalize the discovered dynamics outside the problem scope. We leave the pursuit

of this objective to further research which needs to investigate undercovering the underlying dynamics of a hillslope and fluvial process with physical coefficients and variables rather than elevations of the neighboring nodes.

Secondly, the feature space in our experiment included only the polynomial features. SINDy framework was built taking a secondary objective into account: simplifying the required processing time complexity of its predecessor symbolic regression. This is why SINDy requires the researcher to have a general idea about the dynamic under spotlight in terms of its nature. This prior knowledge enables picking the right category of feature space. We have received positive results utilizing only the polynomial feature space. In contrast, the trigonometric feature space could have also brung suitable information to the output function. This is due to the soil flux processes incorporating the slope between soil elevation of nodes almost all the time. Hence, it would be enlightening to carry out a study which tests the relevancy of the SINDy feature libraries and their combinations on non-linear soil flux processes.

Thirdly, we carried out the study with Landlab simulation data which has a static time-step: 1,250 years. SINDy is designed to infer the dynamical systems with varying time-steps such as moving objects. It incorporates features to differentiate data for different timesteps. Our simulation was carried out with discrete data points with fixed timestep for the sake of simplification of the problem from the perspective of processing time complexity. Though, if SINDy studied the non-linear soil flux processes under varying time steps, it would result in a more comprehensive ability in terms of inference of dynamical system. We leave this problem of demonstration of the performance of SINDy under varying time-steps to further research.

The fourth point that could be considered as a limitation is the weighted scoring we used for the model error determination. The weights were assigned arbitrarily due to processing power limitations instead of more methodical approaches to consider the best weights for different error criteria. For instance, a grid search could be performed to determine the optimal weights. We leave this aspect to future studies.

Finally, it is worth reemphasizing, in the simplest terms, SINDy is a model fitting engine. It does not pledge the explanation of a physical phenomenon deterministically. It is a robust approximation method albeit it does not employ a purely scientific method to explain physical phenomena. This means that it is an efficient apparatus to employ

in potential areas of use we mentioned in the previous section. Above all, it could be considered a gray-box method with some fundamental information and assumptions about the studied dynamical system incorporated to its procedure.

REFERENCES

Baraniuk, R. (2007). Compressive Sensing [Lecture Notes]. *IEEE Signal Processing Magazine*, *24*(4), 118–121. https://doi.org/10.1109/msp.2007.4286571

Bard, J. F. (1974). *Nonlinear parameter estimation*. Academic Press, a subsidiary of Harcourt Brace Jovanovich.

Barnhart, K. R., Glade, R. C., Shobe, C. M., & Tucker, G. E. (2019). Terrainbento 1.0: a Python package for multi-model analysis in long-term drainage basin evolution. *Geoscientific Model Development*, *12*(4), 1267–1297. https://doi.org/10.5194/gmd-12-1267-2019

Bongard, J., & Lipson, H. (2007). Automated reverse engineering of nonlinear dynamical systems. *Proceedings of the National Academy of Sciences of the United States of America*, *104*(24), 9943–9948. https://doi.org/10.1073/pnas.0609476104

Bongard, J., Zykov, V., & Lipson, H. (2006). Resilient machines through continuous Self-Modeling. *Science*, *314*(5802), 1118–1121. https://doi.org/10.1126/science.1133687

Brunton, S. L., Proctor, J. L., & Kutz, J. N. (2016). Discovering governing equations from data by sparse identification of nonlinear dynamical systems. *Proceedings of the National Academy of Sciences of the United States of America*, *113*(15), 3932–3937. https://doi.org/10.1073/pnas.1517384113

Cozad, A., Sahinidis, N. V., & Miller, D. C. (2015). A combined first-principles and data-driven approach to model building. *Computers & Chemical Engineering*, *73*, 116–127. https://doi.org/10.1016/j.compchemeng.2014.11.010

De Prada, C., Hose, D., Gutierrez, G., & Pitarch, J. (2018). DEVELOPING GREY-BOX DYNAMIC PROCESS MODELS. *IFAC-PapersOnLine*, *51*(2), 523–528. https://doi.org/10.1016/j.ifacol.2018.03.088

Hobley, D. E. J., Adams, J. M., Nudurupati, S. S., Hutton, E. W. H., Gasparini, N. M., Istanbulluoglu, E., & Tucker, G. E. (2017). Creative computing with Landlab: an open-source toolkit for building, coupling, and exploring two-dimensional numerical models of Earth-surface dynamics. *Earth Surface Dynamics*, *5*(1), 21–46. https://doi.org/10.5194/esurf-5-21-2017

Holland, J. H. (1992). Genetic Algorithms. *Scientific American*, *267*(1), 66–73. http://www.jstor.org/stable/24939139

Holland, J. H. (1992). Adaptation in natural and artificial systems. In *The MIT Press eBooks*. https://doi.org/10.7551/mitpress/1090.001.0001

Koza, J. (1994a). Genetic programming as a means for programming computers by natural selection. *Statistics and Computing*, *4*(2). https://doi.org/10.1007/bf00175355

Koza, J. (1994b). Genetic programming as a means for programming computers by natural selection. *Statistics and Computing*, *4*(2). https://doi.org/10.1007/bf00175355

Koza, J. R., & Poli, R. (2006). Genetic Programming. In *Springer eBooks* (pp. 127–164). https://doi.org/10.1007/0-387-28356-0_5

Makke, N., & Chawla, S. (2024). Interpretable scientific discovery with symbolic regression: a review. *Artificial Intelligence Review*, *57*(1). https://doi.org/10.1007/s10462-023-10622-0

Schmidt, M., & Lipson, H. (2009). Distilling Free-Form Natural Laws from Experimental Data. *Science*, *324*(5923), 81–85. https://doi.org/10.1126/science.1165893

Tan, K., & Li, Y. (2002). Grey-box model identification via evolutionary computing. *Control Engineering Practice*, *10*(7), 673–684. https://doi.org/10.1016/s0967-0661(02)00031-x

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B. Methodological*, *58*(1), 267–288. https://doi.org/10.1111/j.2517-6161.1996.tb02080.x

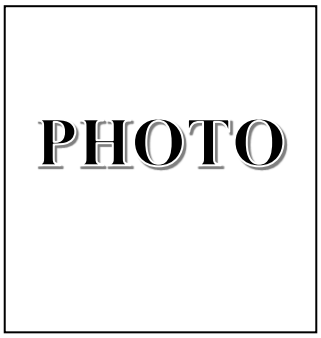
Wang, W. X., Lai, Y., & Grebogi, C. (2016). Data based identification and prediction of nonlinear and complex dynamical systems. *Physics Reports*, *644*, 1–76. https://doi.org/10.1016/j.physrep.2016.06.004

Wishart, D. M. G. (1967). QUEUES AND INVENTORIES. *Journal of the London Mathematical Society*, *s1-42*(1), 382–383. https://doi.org/10.1112/jlms/s1-42.1.382b

APPENDICES

APPENDIX A: Python Jupyter Notebook CD

CURRICULUM VITAE



Name Surname **: Özgür Doğan Birol**

Place and Date of Birth **:**

E-Mail **:**

EDUCATION **:**

- **B.Sc.** **:** 2012, Istanbul Technical University, Management Faculty, Management Engineering Programme
- **M.Sc.** **:** 2022, Istanbul Technical University, Institute of Graduate Studies, Big Data and Data Analytics Programme

PUBLICATIONS, PRESENTATIONS AND PATENTS ON THE THESIS:

- **Birol Ö., Yetemen Ö.** 2024: Sparse Identification of Nonlinear Dynamics (SINDy) of Landscape Evolution Model (LEM) Simulations. International Congress – International Graduate Research Symposium, May 8-10, 2024 Istanbul, Turkey.