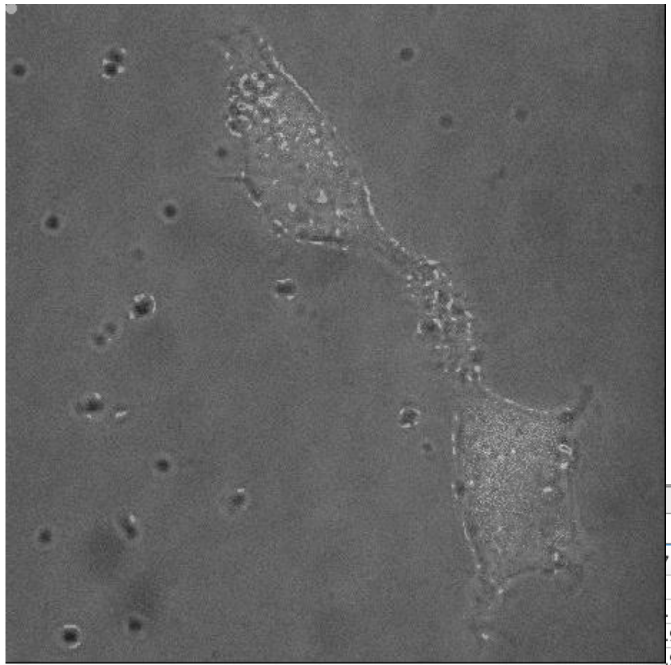
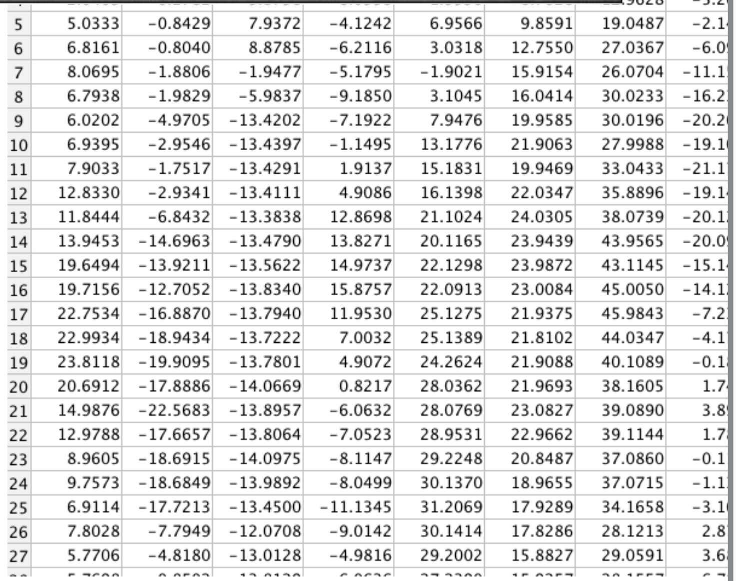
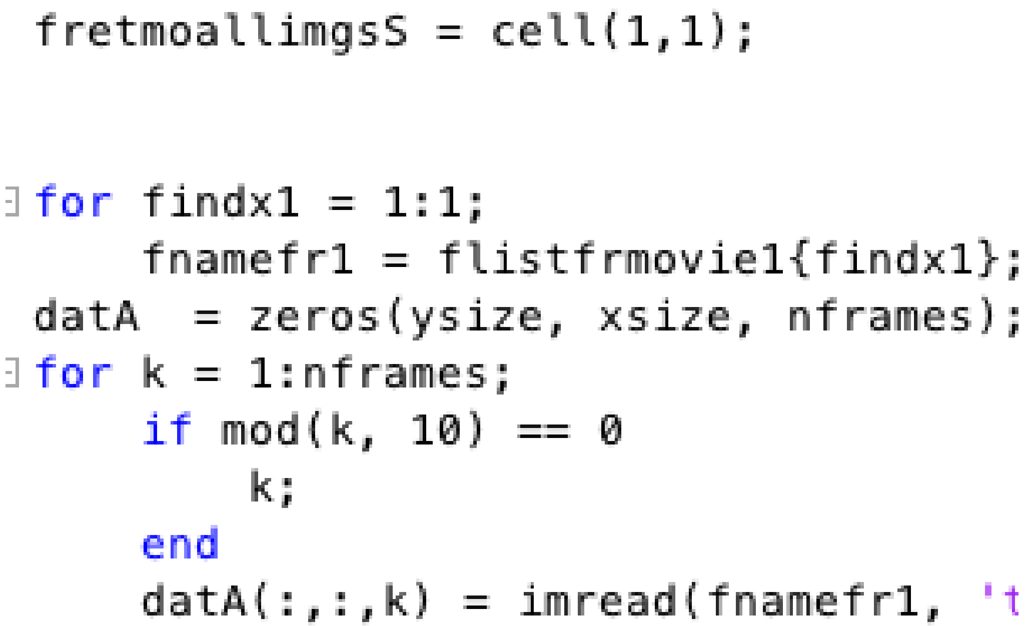
Introduction to Scientific Computation

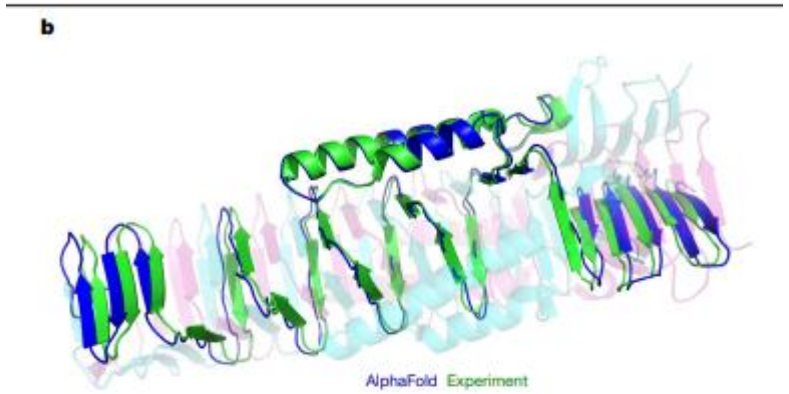


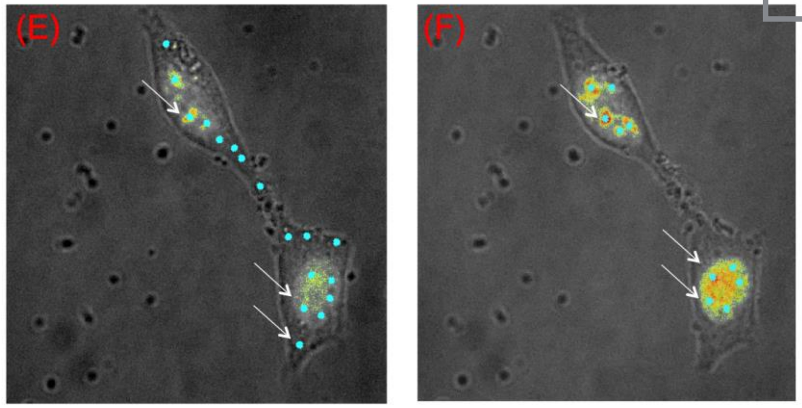


							7	8
							0	
							.1476	-6.0
							.9021	-8.9
							.9628	-5.2
5	5.0333	-0.8429	7.9372	-4.1242	6.9566	9.8591	19.0487	-2.1
6	6.8161	-0.8040	8.8785	-6.2116	3.0318	12.7550	27.0367	-6.0
7	8.0695	-1.8806	-1.9477	-5.1795	-1.9021	15.9154	26.0704	-11.1
8	6.7938	-1.9829	-5.9837	-9.1850	3.1045	16.0414	30.0233	-16.2
9	6.0202	-4.9705	-13.4202	-7.1922	7.9476	19.9585	30.0196	-20.2
10	6.9395	-2.9546	-13.4397	-1.1495	13.1776	21.9063	27.9988	-19.1
11	7.9033	-1.7517	-13.4291	1.9137	15.1831	19.9469	33.0433	-21.1
12	12.8330	-2.9341	-13.4111	4.9086	16.1398	22.0347	35.8896	-19.1
13	11.8444	-6.8432	-13.3838	12.8698	21.1024	24.0305	38.0739	-20.1
14	13.9453	-14.6963	-13.4790	13.8271	20.1165	23.9439	43.9565	-20.0
15	19.6494	-13.9211	-13.5622	14.9737	22.1298	23.9872	43.1145	-15.1
16	19.7156	-12.7052	-13.8340	15.8757	22.0913	23.0084	45.0050	-14.1
17	22.7534	-16.8870	-13.7940	11.9530	25.1275	21.9375	45.9843	-7.2
18	22.9934	-18.9434	-13.7222	7.0032	25.1389	21.8102	44.0347	-4.1
19	23.8118	-19.9095	-13.7801	4.9072	24.2624	21.9088	40.1089	-0.1
20	20.6912	-17.8886	-14.0669	0.8217	28.0362	21.9693	38.1605	1.7
21	14.9876	-22.5683	-13.8957	-6.0632	28.0769	23.0827	39.0890	3.8
22	12.9788	-17.6657	-13.8064	-7.0523	28.9531	22.9662	39.1144	1.7
23	8.9605	-18.6915	-14.0975	-8.1147	29.2248	20.8487	37.0860	-0.1
24	9.7573	-18.6849	-13.9892	-8.0499	30.1370	18.9655	37.0715	-1.1
25	6.9114	-17.7213	-13.4500	-11.1345	31.2069	17.9289	34.1658	-3.1
26	7.8028	-7.7949	-12.0708	-9.0142	30.1414	17.8286	28.1213	2.8
27	5.7706	-4.8180	-13.0128	-4.9816	29.2002	15.8827	29.0591	3.6



```
 fretmoallimgsS = cell(1,1);


for findx1 = 1:1;
    fnamefr1 = flistfrmovie1{findx1};
 datA  = zeros(ysize, xsize, nframes);
for k = 1:nframes;
    if mod(k, 10) == 0
        k;
    end
    datA(:,:,k) = imread(fnamefr1, 't
```





Halil Bayraktar

Lecture 1 - Introduction

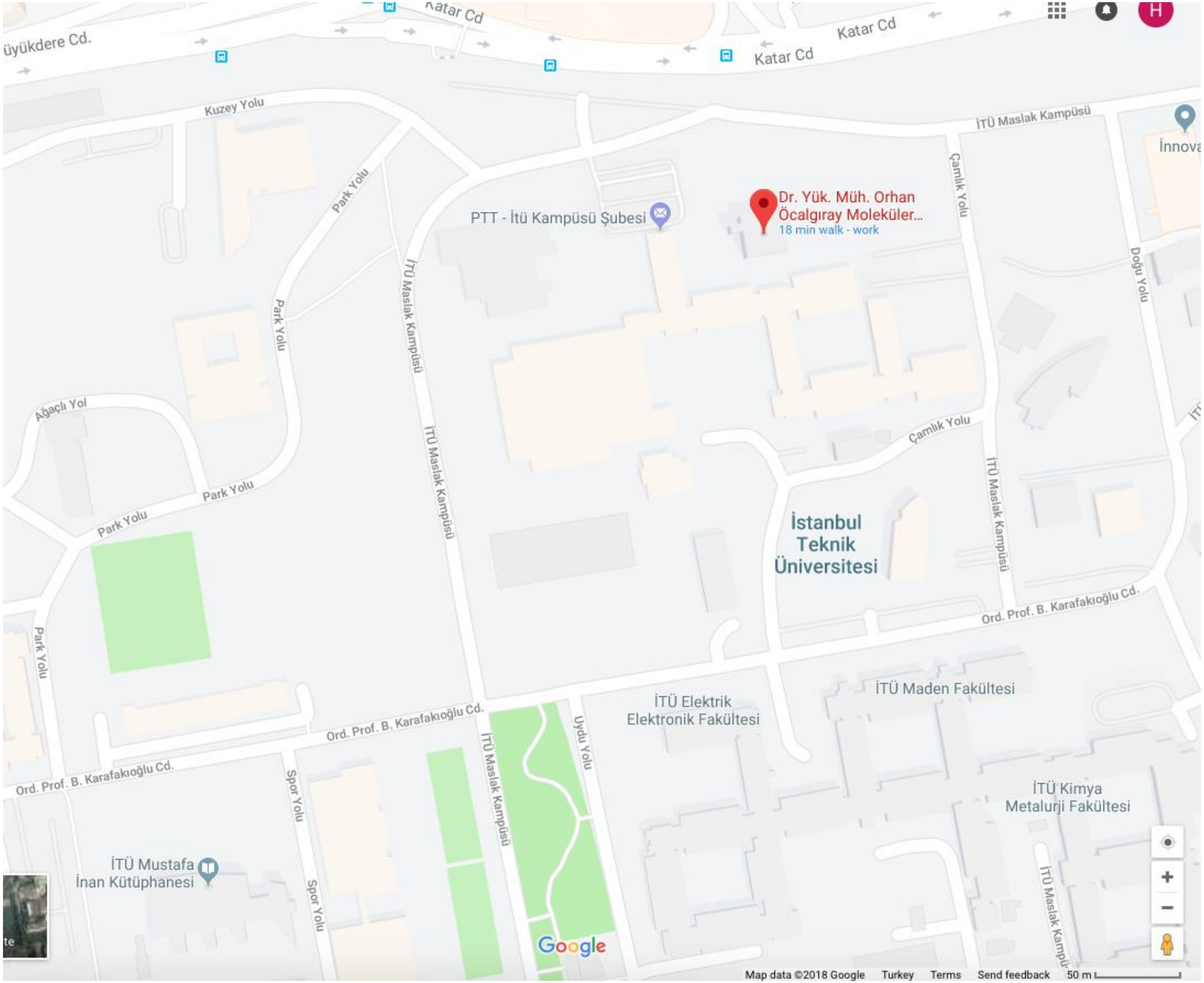
Course Instructor
Halil Bayraktar

Associate Professor
Molecular Biology and Genetic Department
Office : MOBGAM 308

Course Teaching Assistant

Alican Topaloglu
Umut Cagan Ucar

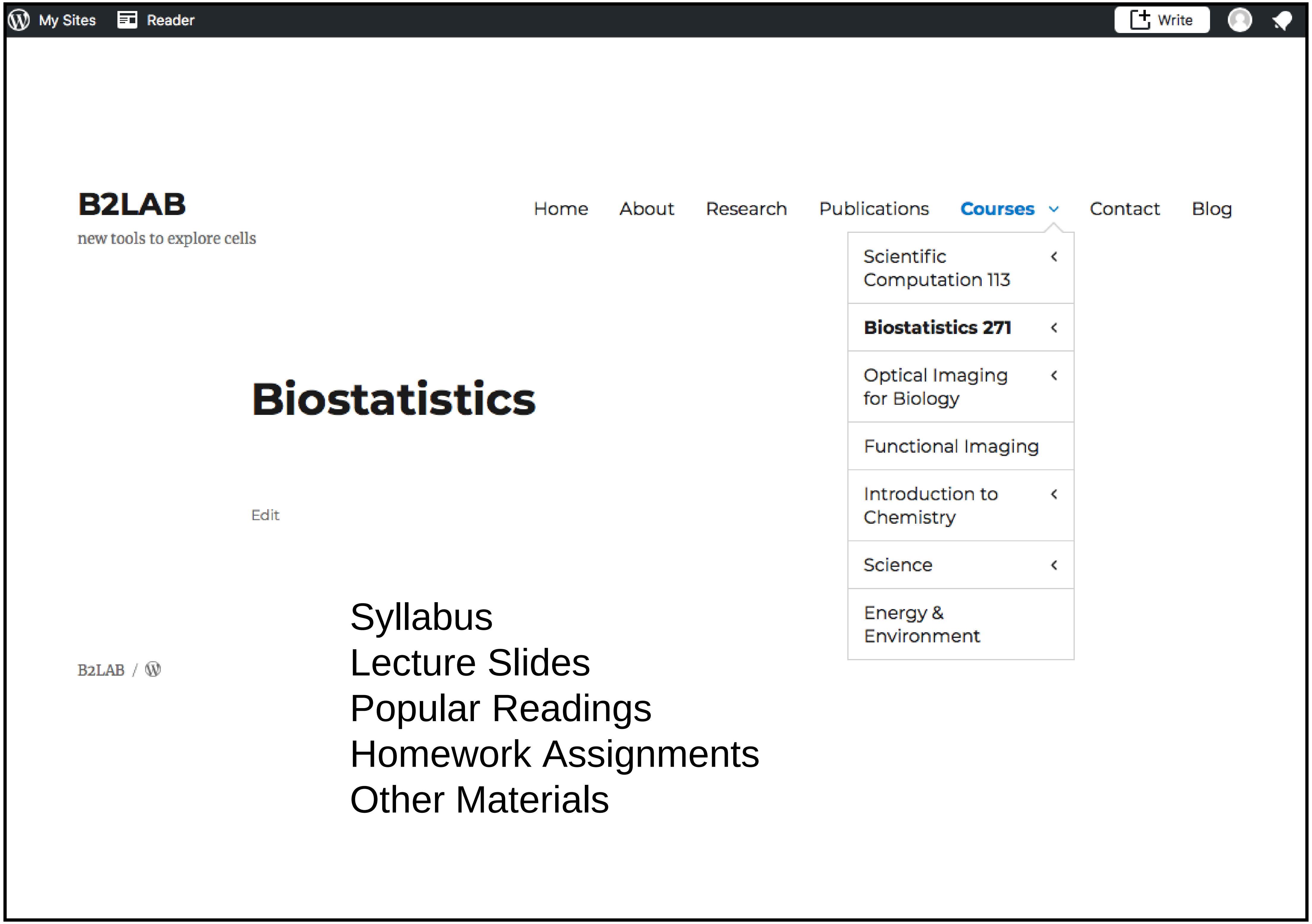
Molecular Biology and Genetics Department Building



Office hours : Monday14:30-15:00 pm and by appointment
hbayraktar@itu.edu.tr

Course website

https://b2lab.wordpress.com/scientific-computing/



B2LAB

new tools to explore cells

Biostatistics

Syllabus
Lecture Slides
Popular Readings
Homework Assignments
Other Materials

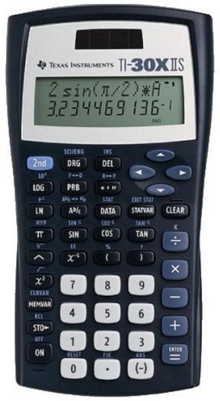
Grading

Percentage	Percentage	Date
Midterm 1	30	to be announced
Midterm 2	30	to be announced
Final	40	to be announced

A minimum average of 20 % from semester activities is required to attend final exam.

Books and other resources

- No official textbook for this course



Please check other reserved books at ITU library and other online resources.

Course Objectives

Basic coding (MATLAB) knowledge to use in your research, other courses and workplace.

Learn coding/algorithm tools to evaluate scientific data

Enhance your data presentation skills for publications (graphs, pie charts, box plots vs.)

Improve your practical mathematics skills

Teaching skills of transition from raw data to presented data

Why study scientific computation?

- Computer and numerical methods are essential to analyze complex data and provide analytical solutions.

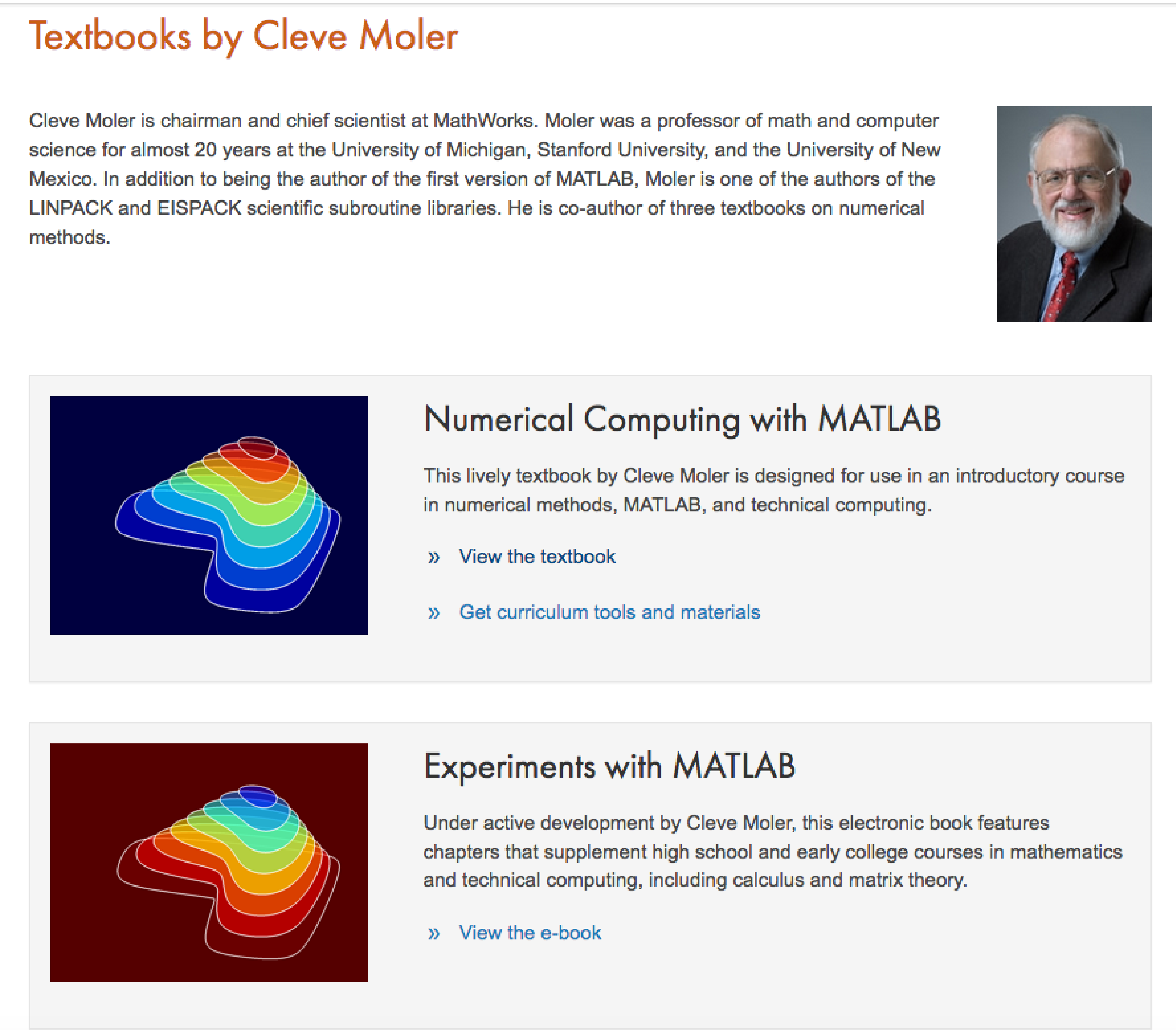
- Presentation of your data with various plots such as line, histogram plot, polar plot, box plot etc.

- Analyze multidimensional data (matrix)
  Example, gene expression profile, microscope images, videos etc.

- Learn to simplify complex problems

- Guidelines to computation

- Learn how to write code to analyze your data



Textbooks by Cleve Moler

Cleve Moler is chairman and chief scientist at MathWorks. Moler was a professor of math and computer science for almost 20 years at the University of Michigan, Stanford University, and the University of New Mexico. In addition to being the author of the first version of MATLAB, Moler is one of the authors of the LINPACK and EISPACK scientific subroutine libraries. He is co-author of three textbooks on numerical methods.

Numerical Computing with MATLAB

This lively textbook by Cleve Moler is designed for use in an introductory course in numerical methods, MATLAB, and technical computing.

» View the textbook

» Get curriculum tools and materials

Experiments with MATLAB

Under active development by Cleve Moler, this electronic book features chapters that supplement high school and early college courses in mathematics and technical computing, including calculus and matrix theory.

» View the e-book

Textbook and other resources

a) Numerical computing with Matlab by Cleve Moler, available for free download at http://www.mathworks.com/moler/.

b) The Matlab demos are available at https://www.mathworks.com/moler/exm/chapters.html

c) Matlab help files

d) Matlab online resources (online videos)

https://www.mathworks.com/matlabcentral/answers/

What can you do with MatLab?

Advance computation/analysis

Data visualization/graphics

Code development

Modeling and simulation

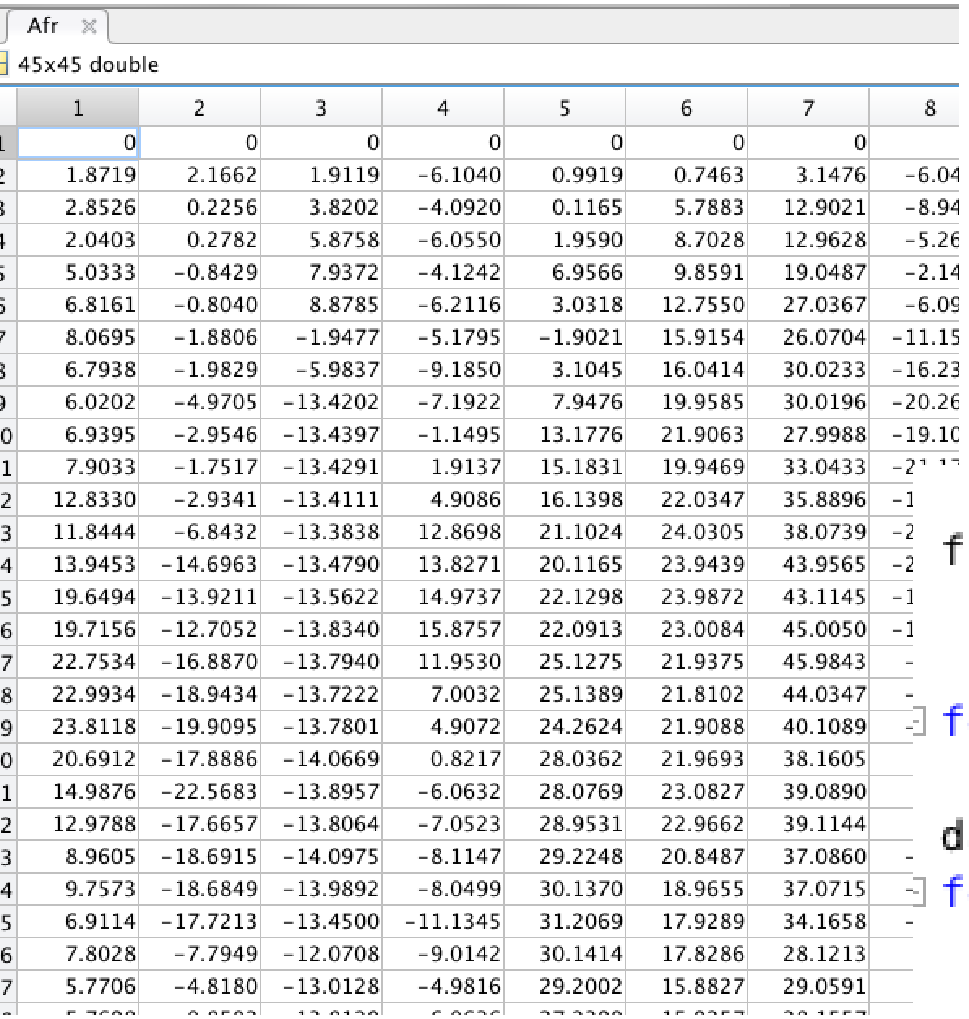
Data acquisition

Graphical user interface

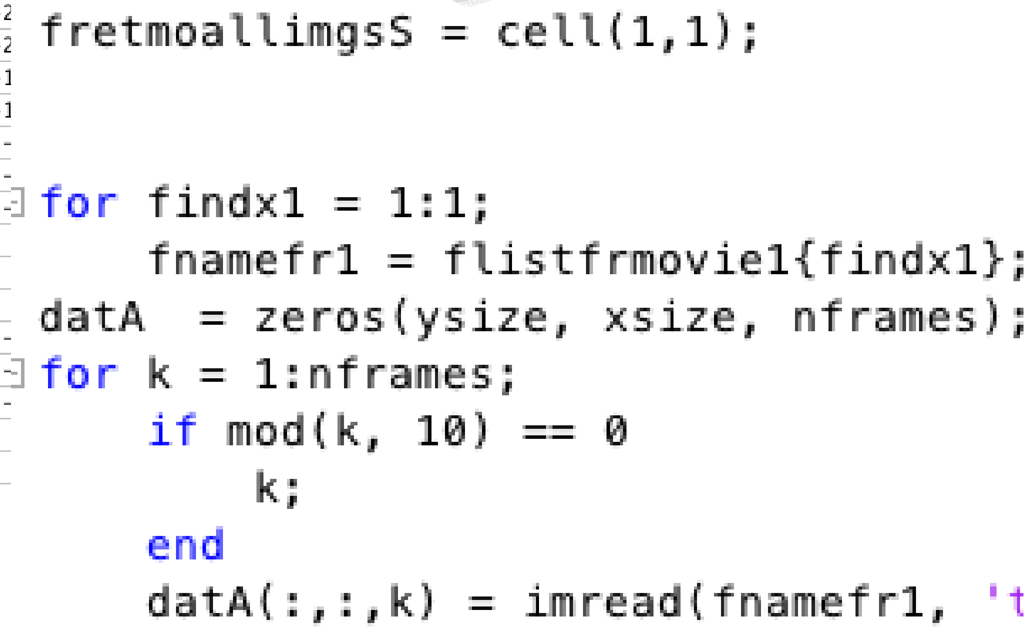
and more…

- Coding for data analysis
- Create your own functions
- Advance plotting
- Create user interfaces (not ideal)
- Interface with other programming languages (JAVA, Python, C/C++)

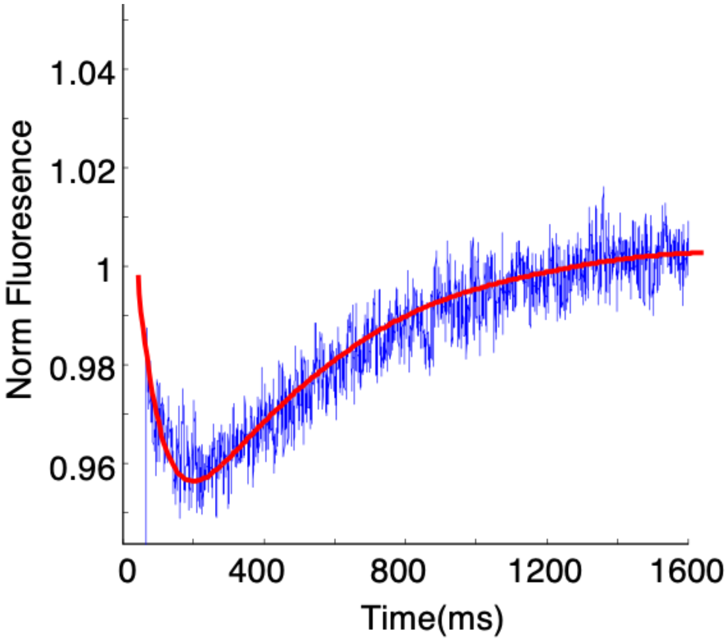
Raw data → Presented data



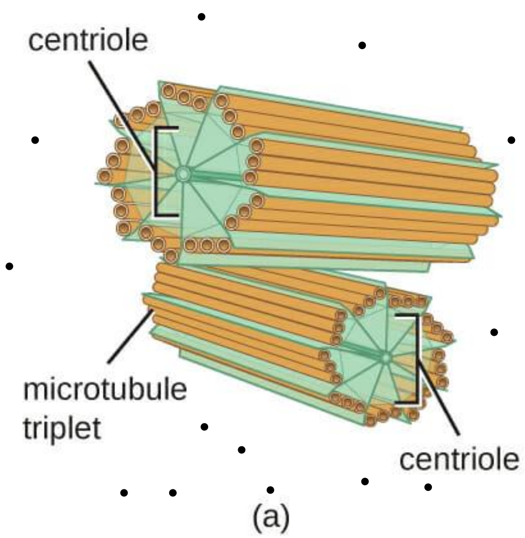
	1	2	3	4	5	6	7	8
1	0	0	0	0	0	0	0	
2	1.8719	2.1662	1.9119	-6.1040	0.9919	0.7463	3.1476	-6.04
3	2.8526	0.2256	3.8202	-4.0920	0.1165	5.7883	12.9021	-8.94
4	2.0403	0.2782	5.8758	-6.0550	1.9590	8.7028	12.9628	-5.26
5	5.0333	-0.8429	7.9372	-4.1242	6.9566	9.8591	19.0487	-2.14
6	6.8161	-0.8040	8.8785	-6.2116	3.0318	12.7550	27.0367	-6.09
7	8.0695	-1.8806	-1.9477	-5.1795	-1.9021	15.9154	26.0704	-11.15
8	6.7938	-1.9829	-5.9837	-9.1850	3.1045	16.0414	30.0233	-16.23
9	6.0202	-4.9705	-13.4202	-7.1922	7.9476	19.9585	30.0196	-20.26
0	6.9395	-2.9546	-13.4397	-1.1495	13.1776	21.9063	27.9988	-19.10
1	7.9033	-1.7517	-13.4291	1.9137	15.1831	19.9469	33.0433	-2
2	12.8330	-2.9341	-13.4111	4.9086	16.1398	22.0347	35.8896	-1
3	11.8444	-6.8432	-13.3838	12.8698	21.1024	24.0305	38.0739	-2
4	13.9453	-14.6963	-13.4790	13.8271	20.1165	23.9439	43.9565	-2
5	19.6494	-13.9211	-13.5622	14.9737	22.1298	23.9872	43.1145	-1
6	19.7156	-12.7052	-13.8340	15.8757	22.0913	23.0084	45.0050	-1
7	22.7534	-16.8870	-13.7940	11.9530	25.1275	21.9375	45.9843	-
8	22.9934	-18.9434	-13.7222	7.0032	25.1389	21.8102	44.0347	-
9	23.8118	-19.9095	-13.7801	4.9072	24.2624	21.9088	40.1089	-
0	20.6912	-17.8886	-14.0669	0.8217	28.0362	21.9693	38.1605	
1	14.9876	-22.5683	-13.8957	-6.0632	28.0769	23.0827	39.0890	
2	12.9788	-17.6657	-13.8064	-7.0523	28.9531	22.9662	39.1144	
3	8.9605	-18.6915	-14.0975	-8.1147	29.2248	20.8487	37.0860	-
4	9.7573	-18.6849	-13.9892	-8.0499	30.1370	18.9655	37.0715	-
5	6.9114	-17.7213	-13.4500	-11.1345	31.2069	17.9289	34.1658	-
6	7.8028	-7.7949	-12.0708	-9.0142	30.1414	17.8286	28.1213	
7	5.7706	-4.8180	-13.0128	-4.9816	29.2002	15.8827	29.0591	

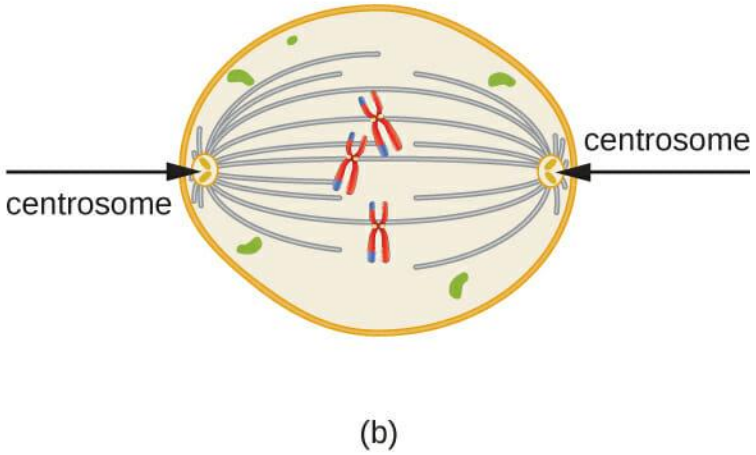


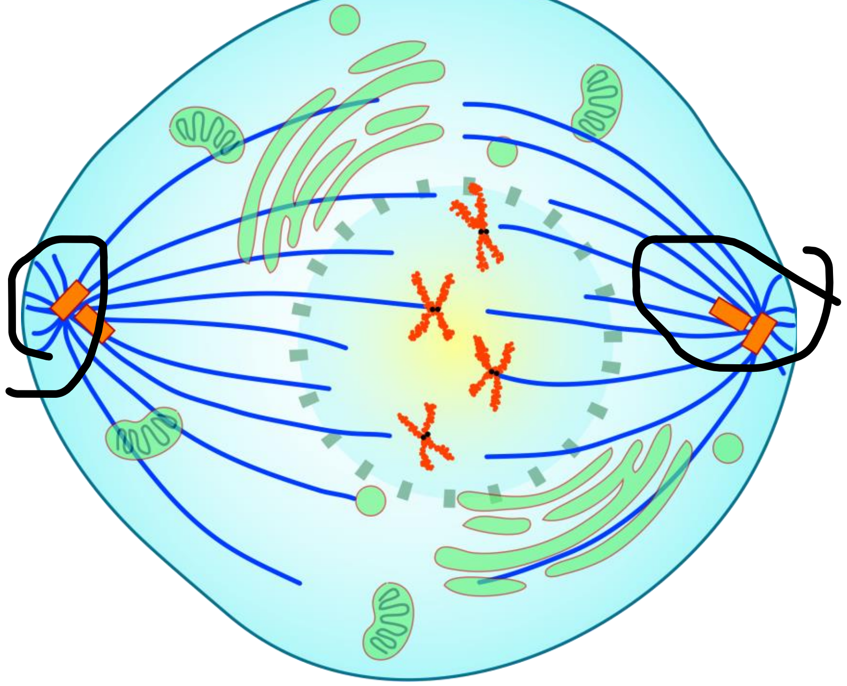
```
fretmoallimgsS = cell(1,1);


for findx1 = 1:1;
    fnamefr1 = flistfrmovie1{findx1};
dataA  = zeros(ysize, xsize, nframes);
for k = 1:nframes;
    if mod(k, 10) == 0
        k;
    end
    dataA(:,:,k) = imread(fnamefr1, 't
```



Scientific computation is a key tool to decipher the mechanism of biology systems.







SCIENTIFIC REPORTS

natureresearch

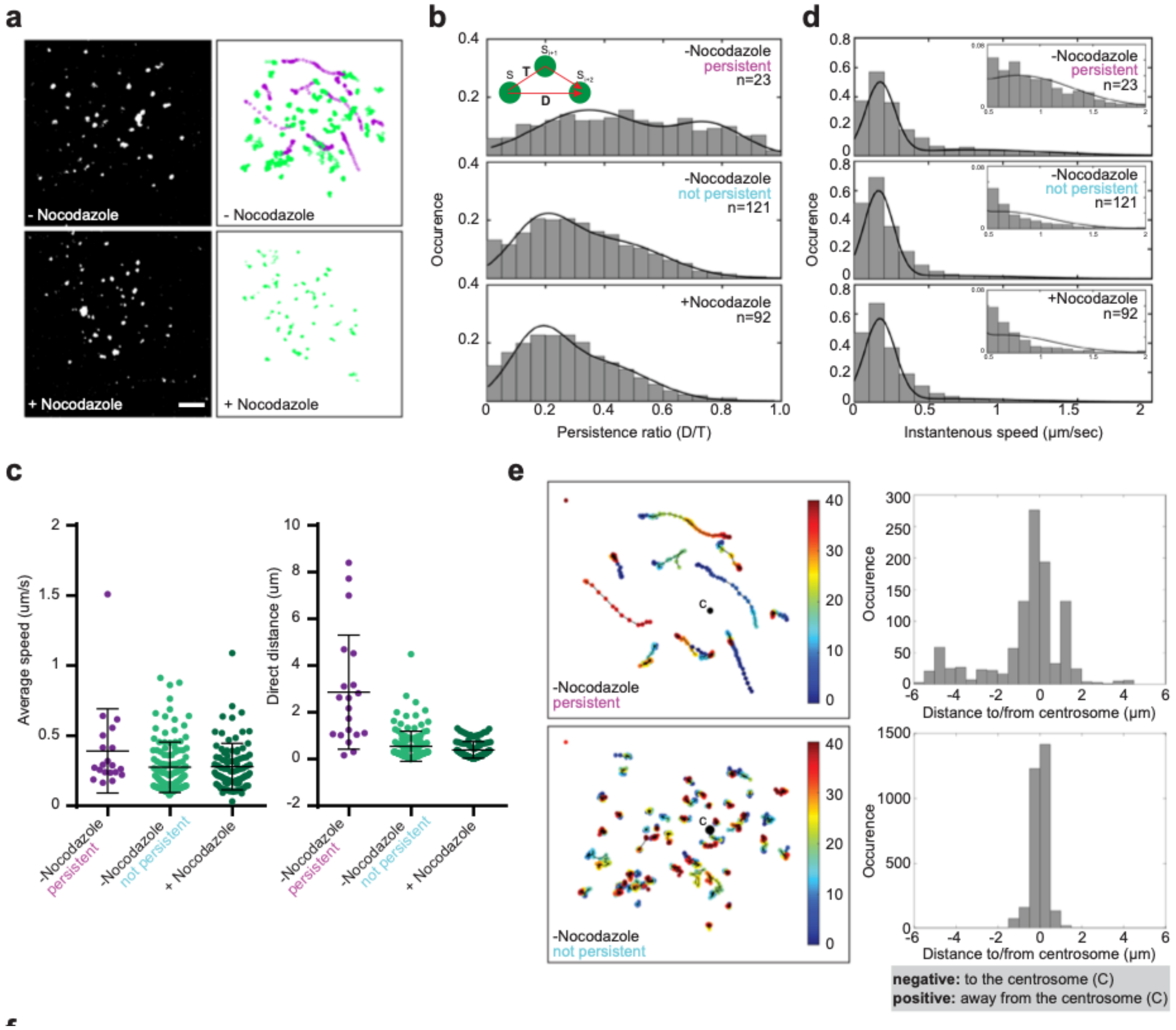
OPEN

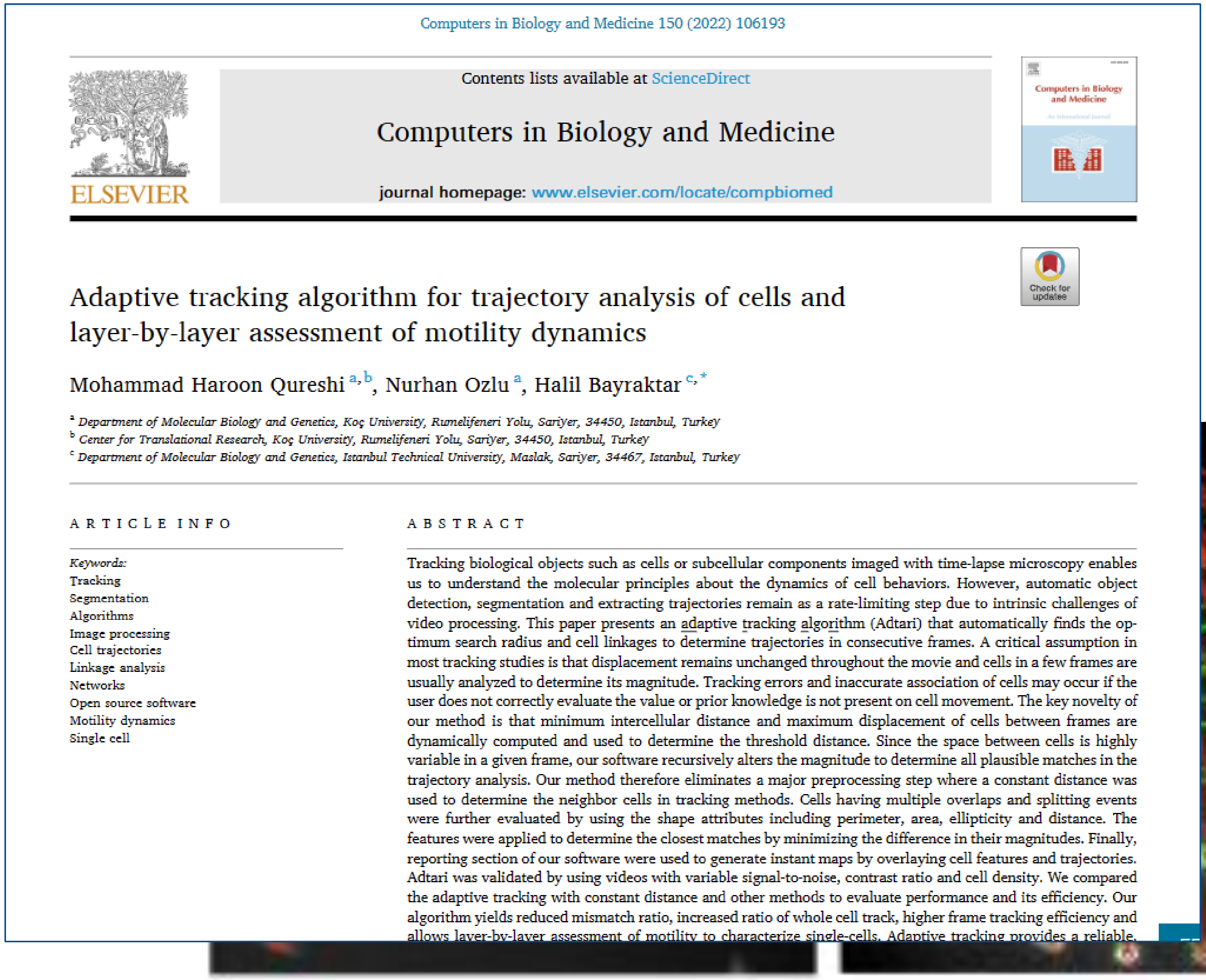
Centrosomal and ciliary targeting of CCDC66 requires cooperative action of centriolar satellites, microtubules and molecular motors

Received: 22 May 2019
Accepted: 11 September 2019
Published online: 03 October 2019

Deniz Conkar[1], Halil Bayraktar[2] & Elif Nur Firat-Karalar[1]

Mammalian centrosomes and cilia play key roles in many cellular processes and their deregulation is linked to cancer and ciliopathies. Spatiotemporal regulation of their biogenesis and function in response to physiological stimuli requires timely protein targeting. This can occur by different pathways, including microtubule-dependent active transport and via centriolar satellites, which are key regulators of cilia assembly and signaling. How satellites mediate their functions and their relationship with other targeting pathways is currently unclear. To address this, we studied retinal degeneration gene product CCDC66, which localizes to centrosomes, cilia, satellites and microtubules and functions in ciliogenesis. FRAP experiments showed that its centrosomal pool was dynamic and the ciliary pool associated with the ciliary axoneme and was stable. Centrosomal CCDC66 abundance and dynamics required microtubule-dependent active transport and tethering, and was inhibited by sequestration at satellites. Systematic quantitation of satellite dynamics identified only a small fraction to display microtubule-based bimodal motility, consistent with trafficking function. Majority displayed diffusive motility with unimodal persistence, supporting sequestration function. Together, our findings reveal new mechanisms of communication between membrane-less compartments.





Computers in Biology and Medicine 150 (2022) 106193

Contents lists available at ScienceDirect

Computers in Biology and Medicine

journal homepage: www.elsevier.com/locate/compbiomed

Adaptive tracking algorithm for trajectory analysis of cells and layer-by-layer assessment of motility dynamics

Mohammad Haroon Qureshi[a,b], Nurhan Ozlu[a], Halil Bayraktar[c,*]

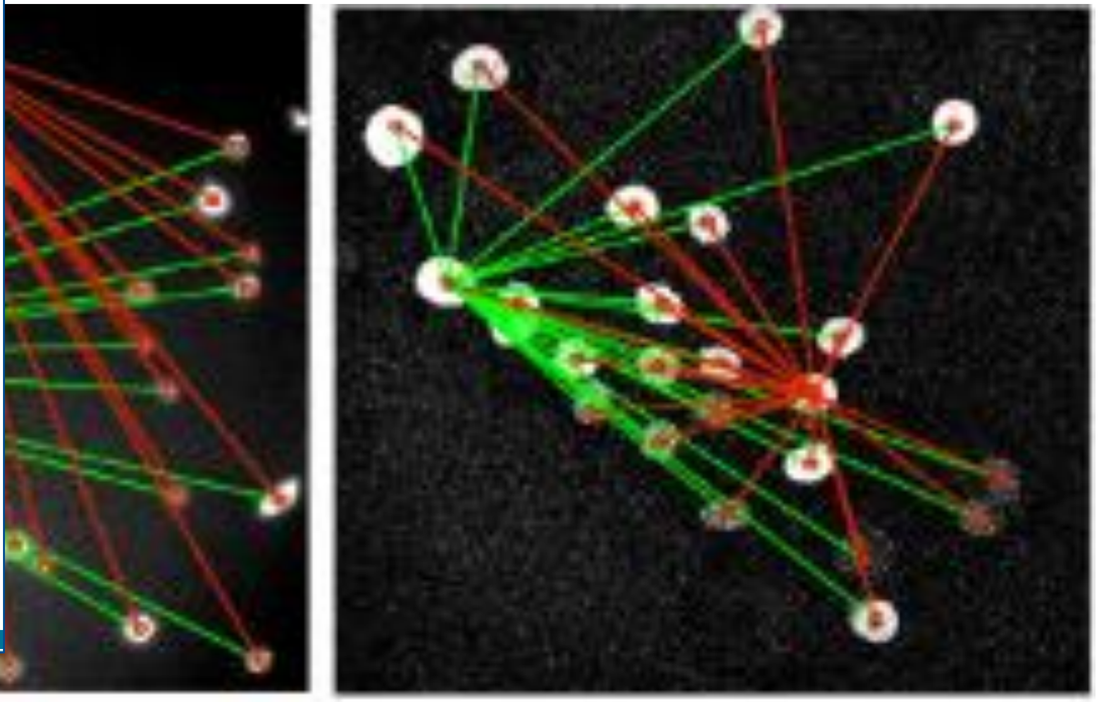
[a] Department of Molecular Biology and Genetics, Koç University, Rumelifeneri Yolu, Sarıyer, 34450, Istanbul, Turkey
[b] Center for Translational Research, Koç University, Rumelifeneri Yolu, Sarıyer, 34450, Istanbul, Turkey
[c] Department of Molecular Biology and Genetics, Istanbul Technical University, Maslak, Sarıyer, 34467, Istanbul, Turkey

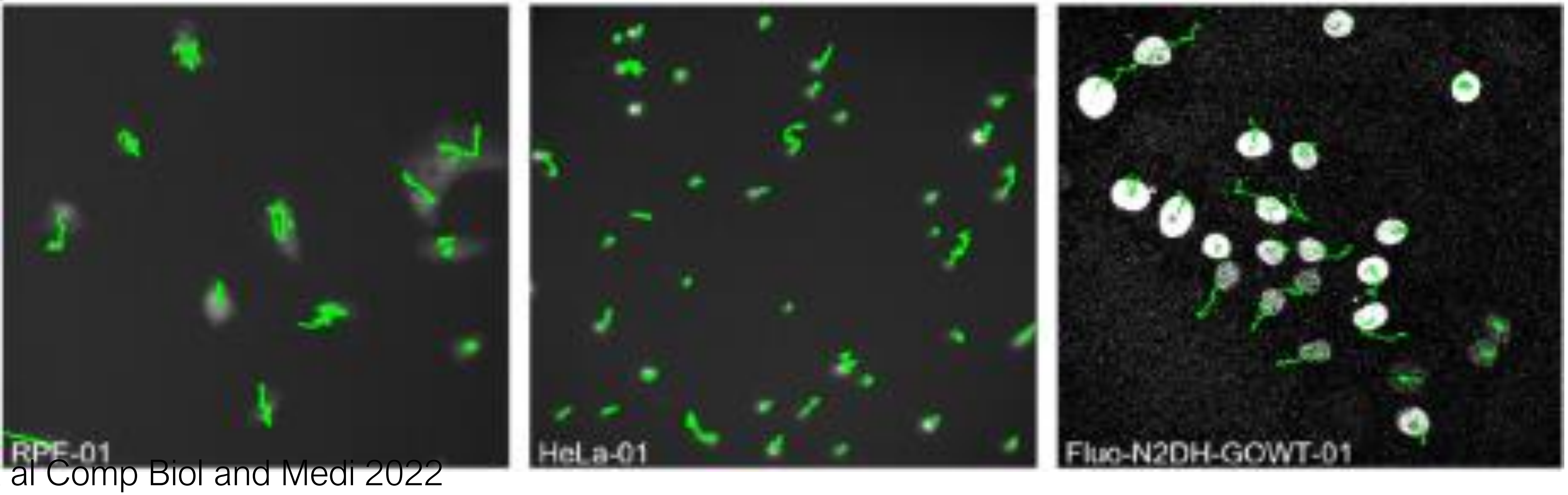
ARTICLE INFO

Keywords:
Tracking
Segmentation
Algorithms
Image processing
Cell trajectories
Linkage analysis
Networks
Open source software
Motility dynamics
Single cell

ABSTRACT

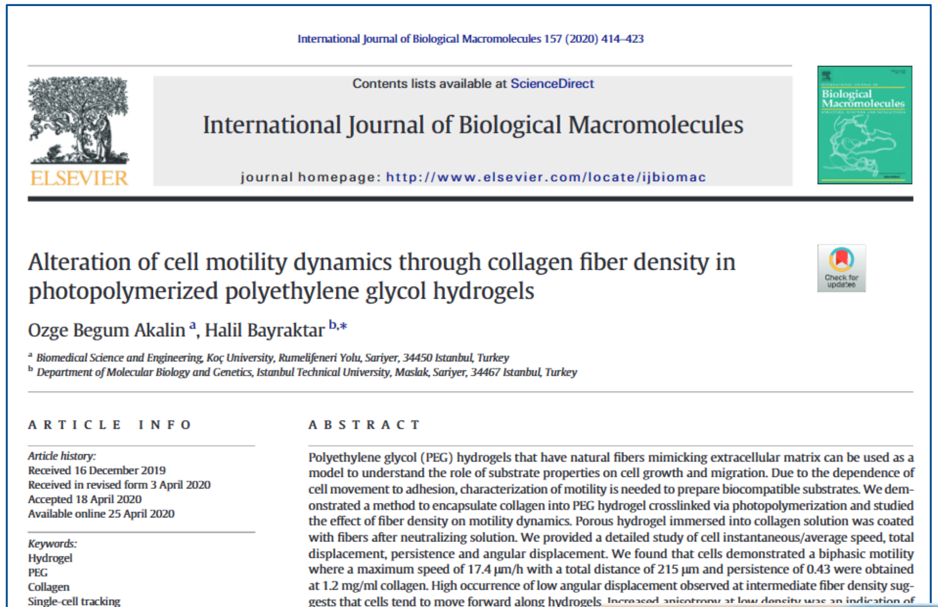
Tracking biological objects such as cells or subcellular components imaged with time-lapse microscopy enables us to understand the molecular principles about the dynamics of cell behaviors. However, automatic object detection, segmentation and extracting trajectories remain as a rate-limiting step due to intrinsic challenges of video processing. This paper presents an adaptive tracking algorithm (Adtari) that automatically finds the optimum search radius and cell linkages to determine trajectories in consecutive frames. A critical assumption in most tracking studies is that displacement remains unchanged throughout the movie and cells in a few frames are usually analyzed to determine its magnitude. Tracking errors and inaccurate association of cells may occur if the user does not correctly evaluate the value or prior knowledge is not present on cell movement. The key novelty of our method is that minimum intercellular distance and maximum displacement of cells between frames are dynamically computed and used to determine the threshold distance. Since the space between cells is highly variable in a given frame, our software recursively alters the magnitude to determine all plausible matches in the trajectory analysis. Our method therefore eliminates a major preprocessing step where a constant distance was used to determine the neighbor cells in tracking methods. Cells having multiple overlaps and splitting events were further evaluated by using the shape attributes including perimeter, area, ellipticity and distance. The features were applied to determine the closest matches by minimizing the difference in their magnitudes. Finally, reporting section of our software were used to generate instant maps by overlaying cell features and trajectories. Adtari was validated by using videos with variable signal-to-noise, contrast ratio and cell density. We compared the adaptive tracking with constant distance and other methods to evaluate performance and its efficiency. Our algorithm yields reduced mismatch ratio, increased ratio of whole cell track, higher frame tracking efficiency and allows layer-by-layer assessment of motility to characterize single-cells. Adaptive tracking provides a reliable





Qureshi et al Comp Biol and Medi 2022

Material science and computation: Adaptive tracking to study cell motility at PEG-collagen hydrogels



International Journal of Biological Macromolecules 157 (2020) 414–423

Contents lists available at ScienceDirect

International Journal of Biological Macromolecules

journal homepage: http://www.elsevier.com/locate/ijbiomac

Alteration of cell motility dynamics through collagen fiber density in photopolymerized polyethylene glycol hydrogels

Ozge Begum Akalin [a], Halil Bayraktar [b,*]

[a] Biomedical Science and Engineering, Koç University, Rumelifeneri Yolu, Sariyer, 34450 Istanbul, Turkey
[b] Department of Molecular Biology and Genetics, Istanbul Technical University, Maslak, Sariyer, 34467 Istanbul, Turkey

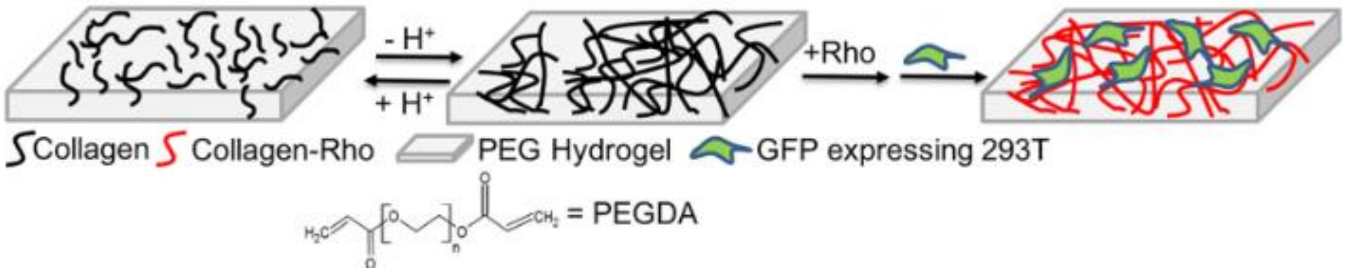
ARTICLE INFO

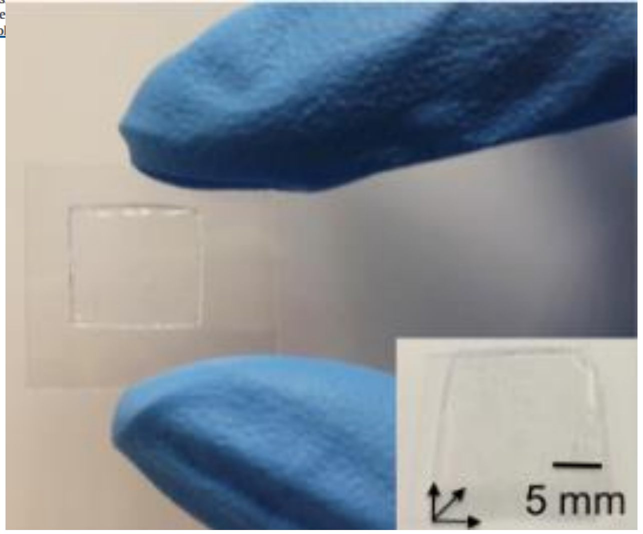
Article history:
Received 16 December 2019
Received in revised form 3 April 2020
Accepted 18 April 2020
Available online 25 April 2020

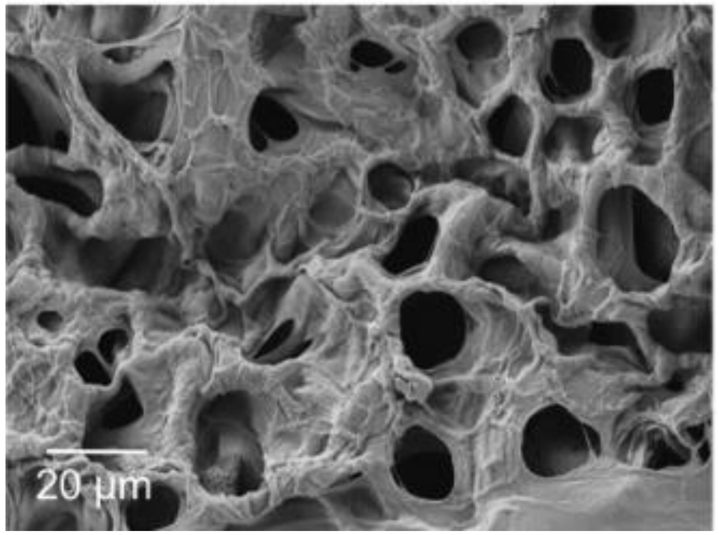
Keywords:
Hydrogel
PEG
Collagen
Single-cell tracking
Motility dynamics
Persistence

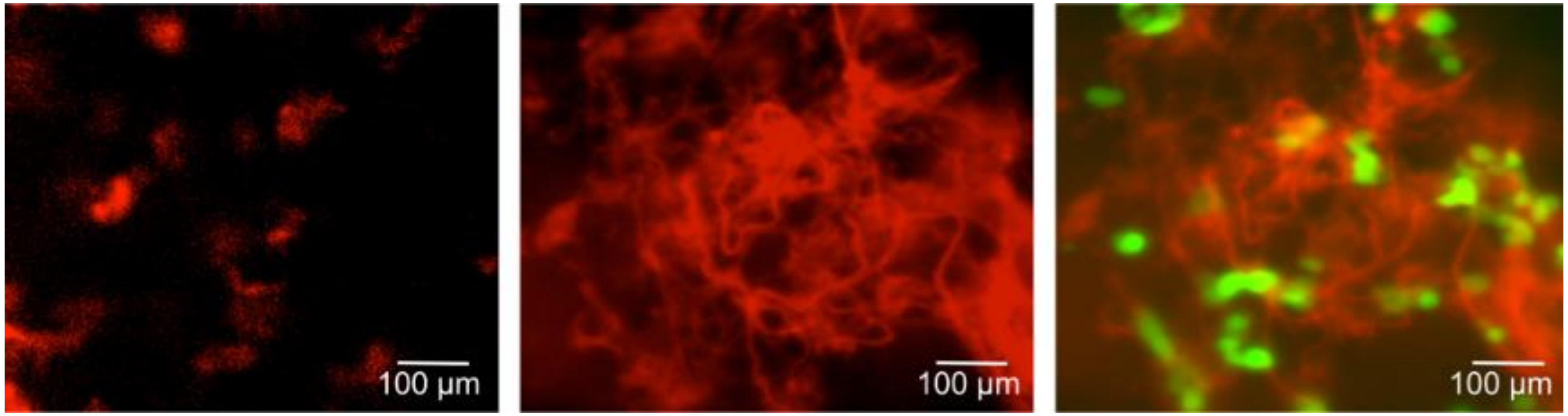
ABSTRACT

Polyethylene glycol (PEG) hydrogels that have natural fibers mimicking extracellular matrix can be used as a model to understand the role of substrate properties on cell growth and migration. Due to the dependence of cell movement to adhesion, characterization of motility is needed to prepare biocompatible substrates. We demonstrated a method to encapsulate collagen into PEG hydrogel crosslinked via photopolymerization and studied the effect of fiber density on motility dynamics. Porous hydrogel immersed into collagen solution was coated with fibers after neutralizing solution. We provided a detailed study of cell instantaneous/average speed, total displacement, persistence and angular displacement. We found that cells demonstrated a biphasic motility where a maximum speed of 17.4 μm/h with a total distance of 215 μm and persistence of 0.43 were obtained at 1.2 mg/ml collagen. High occurrence of low angular displacement observed at intermediate fiber density suggests that cells tend to move forward along hydrogels. Increased anisotropy at low density was an indication of forward and backward movement. Finally, matrix de
beads by tracking fiber displacement at subpixel resol









Machine Learning and AI

- Supervised learning: Build a model based on input and output data
- Unsupervised learning: Build a model based on input data

Supervised learning uses classification and regression tech to develop a model
Example,
Can a model predict the diving and nondiving cells in images?
Can a model predict the negative or positive feedbacks from a text message?
Can a model predict 3D structure of proteins?
Can a model discover a new drug?

Process of learning
- Collection: Gathering data
- Data Preprocessing: Cleaning and preparing the data for machine learning algorithms.
- Model Training: Building a model by fitting it to the prepared data.
- Model Evaluation: Assessing the performance of the trained model on unseen data.
- Prediction: Using the trained model to make predictions on new data.

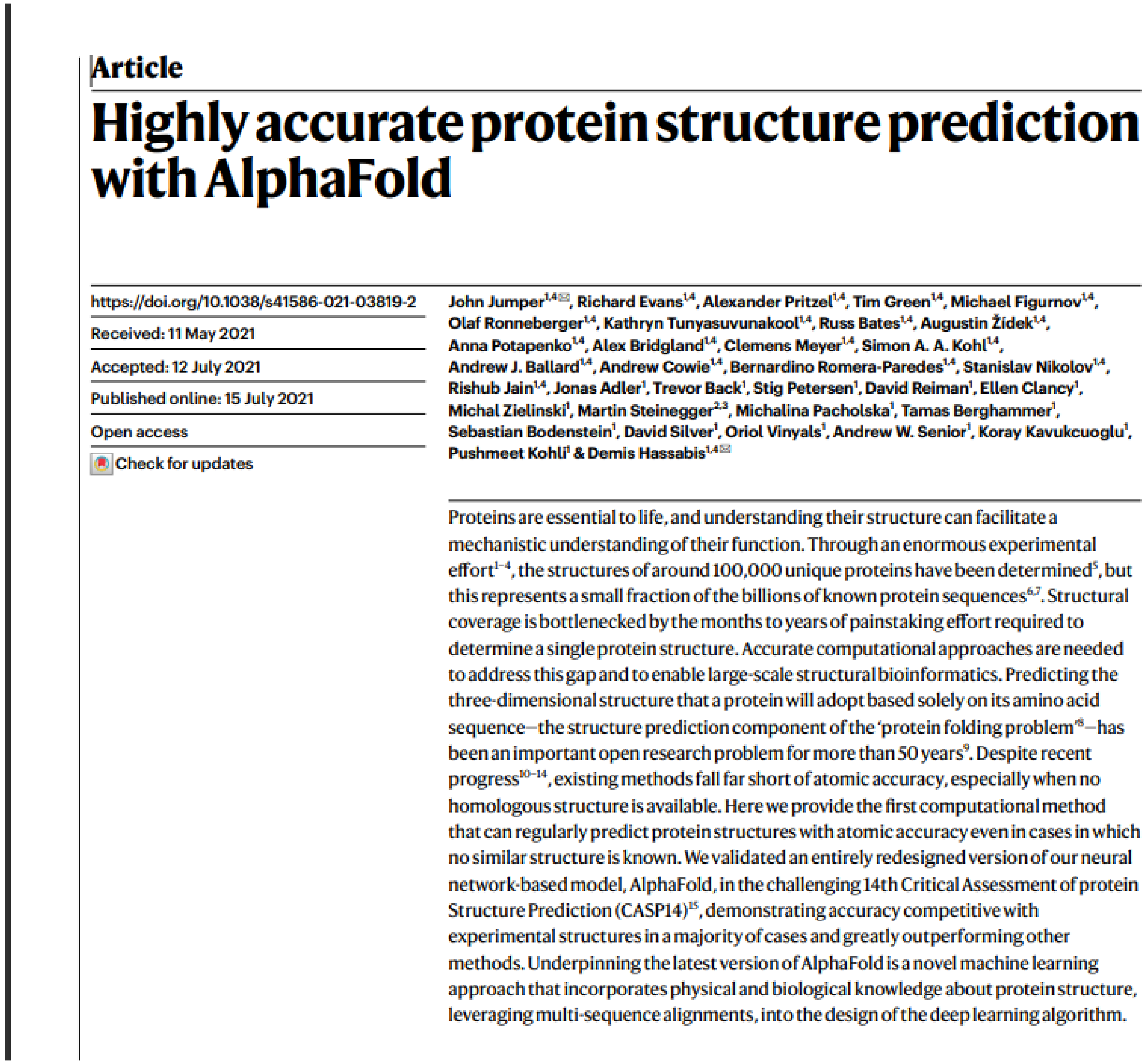
Protein Structure prediction with AI

AlphaFold developed by deepmind

It precits the protein’s #D structure based on its amino acid sequence

AlphaFold Protein Structure Database (ebi.ac.uk)

AlphaFold - Google DeepMind



Article

Highly accurate protein structure prediction with AlphaFold

https://doi.org/10.1038/s41586-021-03819-2

Received: 11 May 2021

Accepted: 12 July 2021

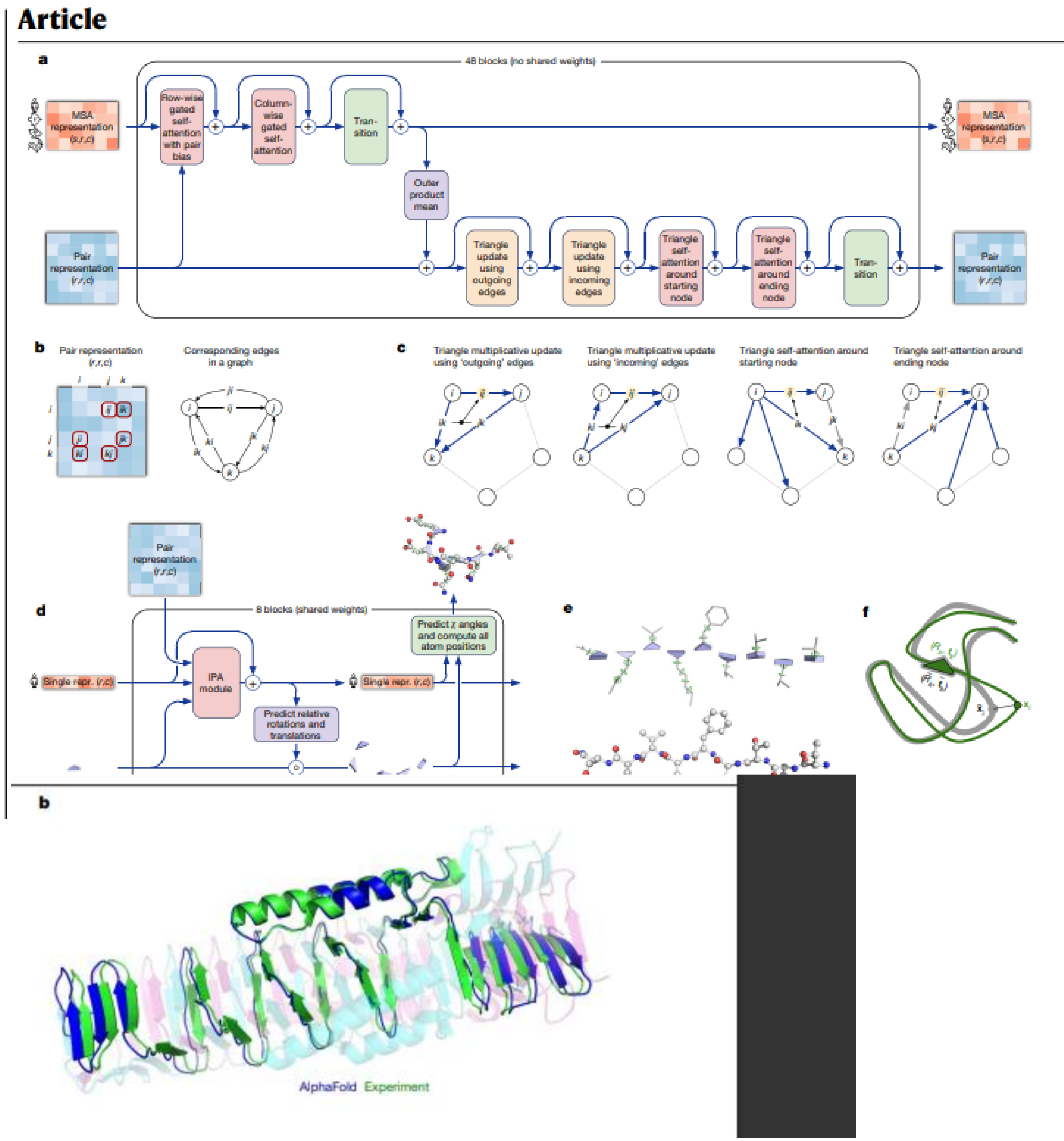
Published online: 15 July 2021

Open access

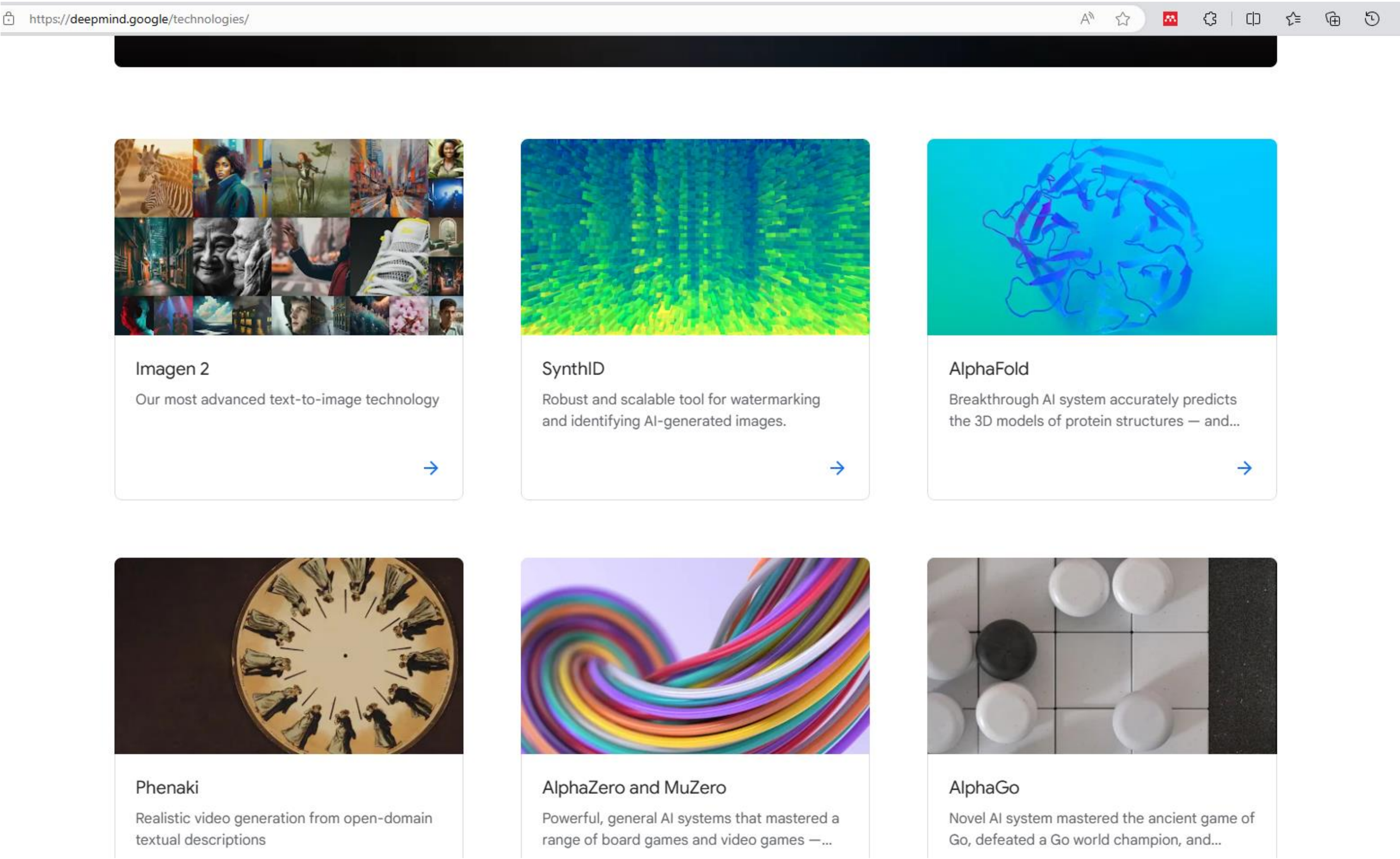
Check for updates

John Jumper[1,4]✉, Richard Evans[1,4], Alexander Pritzel[1,4], Tim Green[1,4], Michael Figurnov[1,4], Olaf Ronneberger[1,4], Kathryn Tunyasuvunakool[1,4], Russ Bates[1,4], Augustin Žídek[1,4], Anna Potapenko[1,4], Alex Bridgland[1,4], Clemens Meyer[1,4], Simon A. A. Kohl[1,4], Andrew J. Ballard[1,4], Andrew Cowie[1,4], Bernardino Romera-Paredes[1,4], Stanislav Nikolov[1,4], Rishub Jain[1,4], Jonas Adler[1], Trevor Back[1], Stig Petersen[1], David Reiman[1], Ellen Clancy[1], Michal Zielinski[1], Martin Steinegger[2,3], Michalina Pacholska[1], Tamas Berghammer[1], Sebastian Bodenstein[1], David Silver[1], Oriol Vinyals[1], Andrew W. Senior[1], Koray Kavukcuoglu[1], Pushmeet Kohli[1] & Demis Hassabis[1,4]✉

Proteins are essential to life, and understanding their structure can facilitate a mechanistic understanding of their function. Through an enormous experimental effort[1–4], the structures of around 100,000 unique proteins have been determined[5], but this represents a small fraction of the billions of known protein sequences[6,7]. Structural coverage is bottlenecked by the months to years of painstaking effort required to determine a single protein structure. Accurate computational approaches are needed to address this gap and to enable large-scale structural bioinformatics. Predicting the three-dimensional structure that a protein will adopt based solely on its amino acid sequence—the structure prediction component of the ‘protein folding problem’[8]—has been an important open research problem for more than 50 years[9]. Despite recent progress[10–14], existing methods fall far short of atomic accuracy, especially when no homologous structure is available. Here we provide the first computational method that can regularly predict protein structures with atomic accuracy even in cases in which no similar structure is known. We validated an entirely redesigned version of our neural network-based model, AlphaFold, in the challenging 14th Critical Assessment of protein Structure Prediction (CASP14)[15], demonstrating accuracy competitive with experimental structures in a majority of cases and greatly outperforming other methods. Underpinning the latest version of AlphaFold is a novel machine learning approach that incorporates physical and biological knowledge about protein structure, leveraging multi-sequence alignments, into the design of the deep learning algorithm.


Article
a
48 blocks (no shared weights)
MSA representation (s,r,c)
Row-wise gated self-attention with pair bias
Column-wise gated self-attention
Transition
Outer product mean
Pair representation (r,r,c)
Triangle update using outgoing edges
Triangle update using incoming edges
Triangle self-attention around starting node
Triangle self-attention around ending node
Transition
MSA representation (s,r,c)
Pair representation (r,r,c)
b
Pair representation (r,r,c)
Corresponding edges in a graph
c
Triangle multiplicative update using 'outgoing' edges
Triangle multiplicative update using 'incoming' edges
Triangle self-attention around starting node
Triangle self-attention around ending node
i
j
k
ij
ik
ji
jk
ki
kj
d
Pair representation (r,r,c)
8 blocks (shared weights)
Single repr. (r,c)
IPA module
Predict relative rotations and translations
Single repr. (r,c)
Predict χ angles and compute all atom positions
e
f
b
AlphaFold Experiment

Google deepmind



https://deepmind.google/technologies/

Imagen 2

Our most advanced text-to-image technology

SynthID

Robust and scalable tool for watermarking and identifying AI-generated images.

AlphaFold

Breakthrough AI system accurately predicts the 3D models of protein structures — and...

Phenaki

Realistic video generation from open-domain textual descriptions

AlphaZero and MuZero

Powerful, general AI systems that mastered a range of board games and video games —...

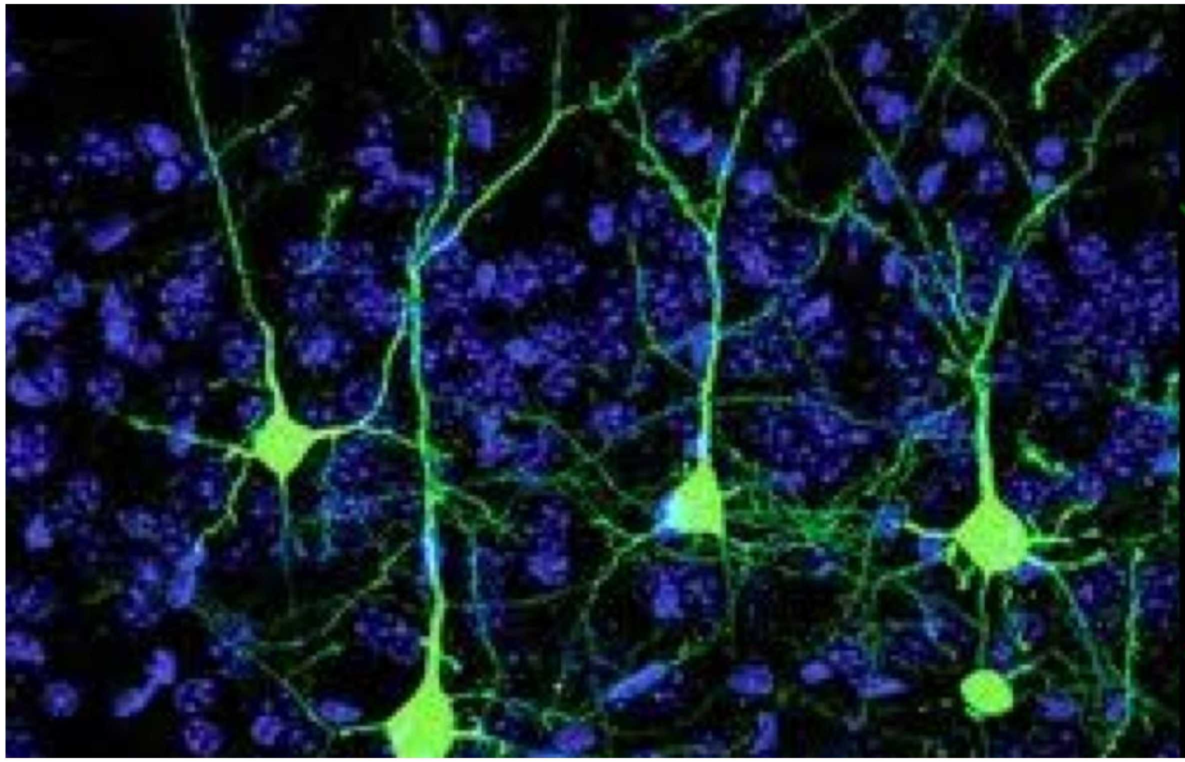
AlphaGo

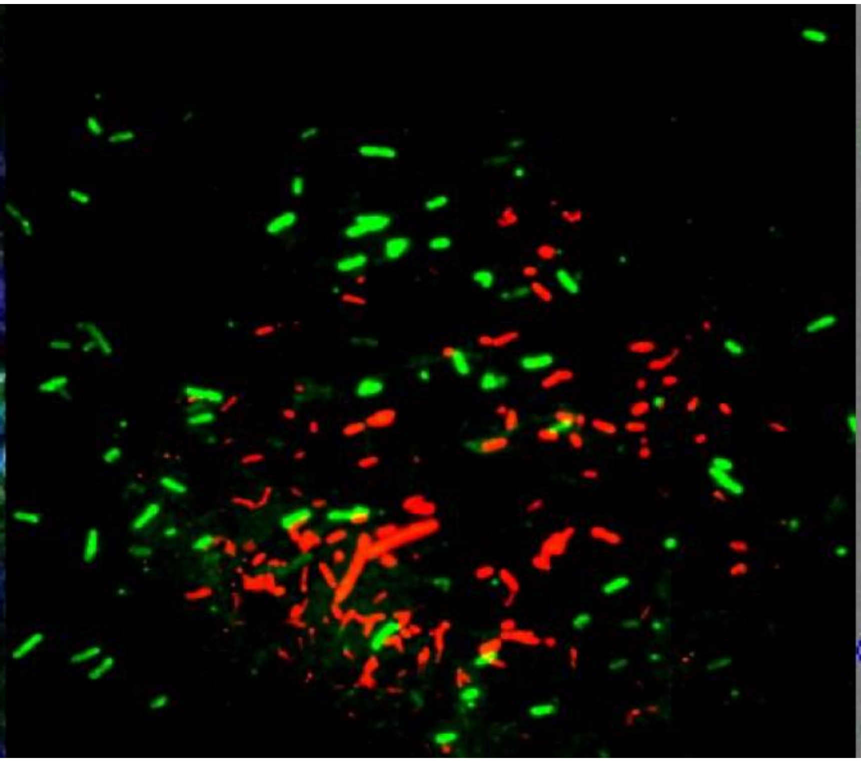
Novel AI system mastered the ancient game of Go, defeated a Go world champion, and...

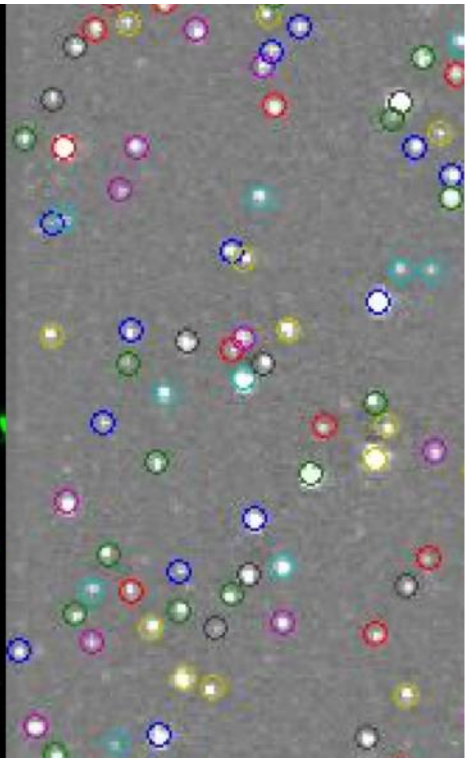
Application of ML

- Image recognition: Face recognition, medical image, self-driving cars.
- Natural language processing: Text analytics, sentiment analysis.
- Recommendation systems: Product recommendations, music recommendations, movie recommendations.
- Banking: Identifying fraudulent transactions, credit score.
- Protein folding
- Drug discovery

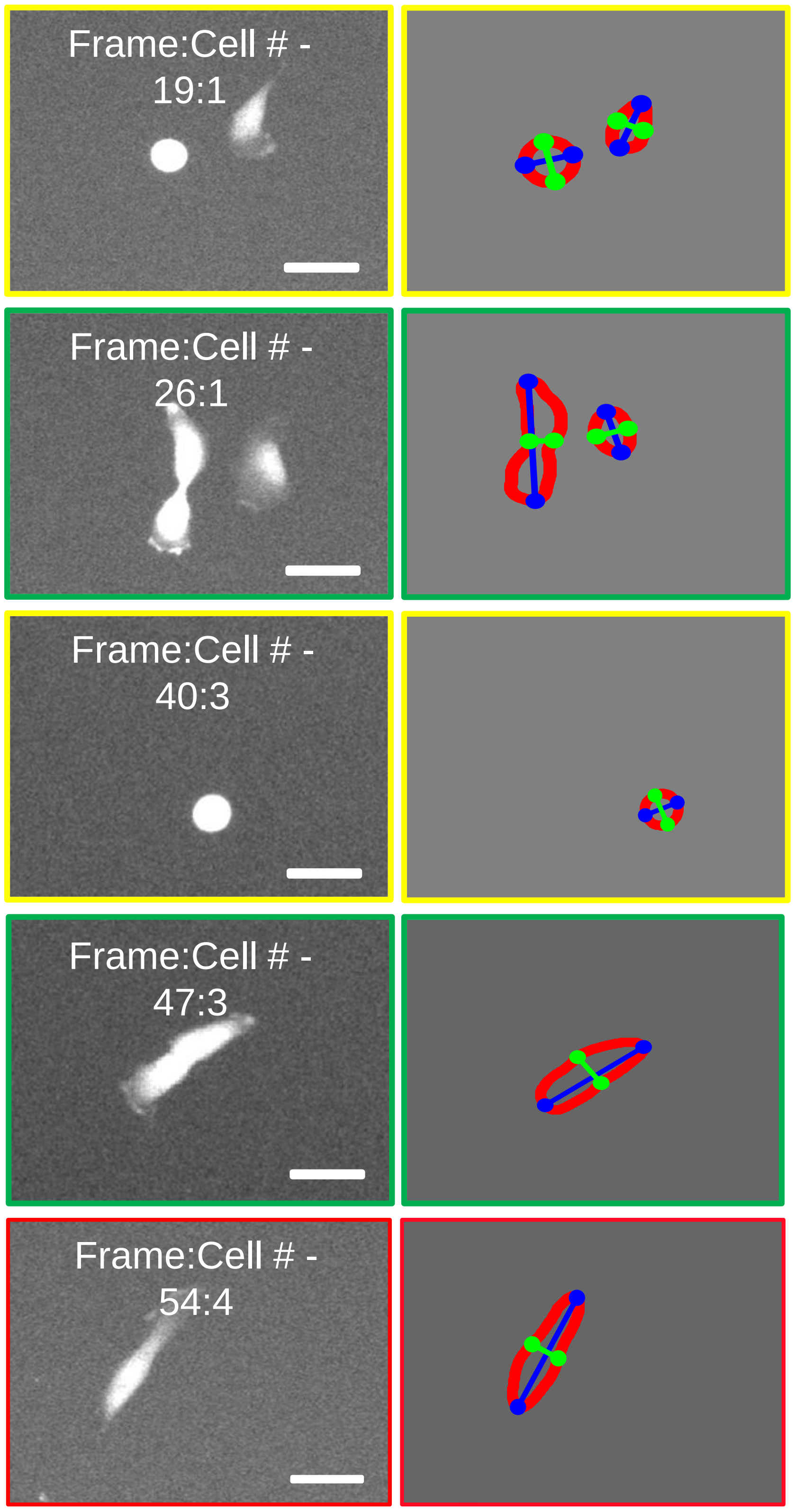
How many cell, bacteria or single molecule is present in each picture?

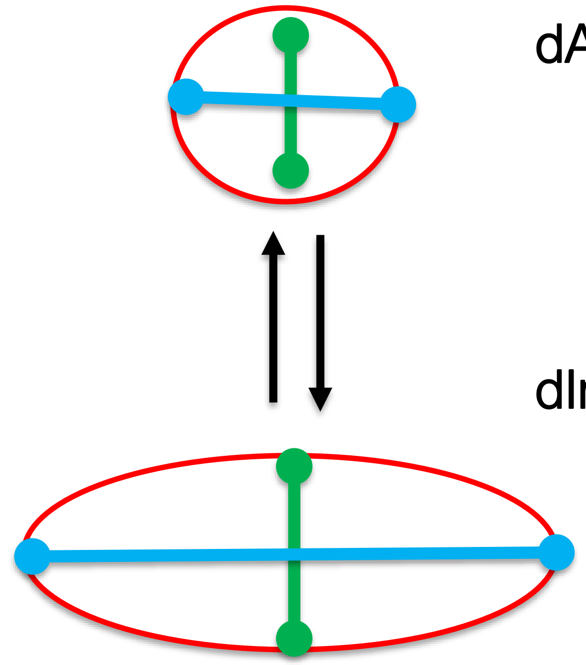






Time course of geometric events: event map generation for identification of cell splitting





Minor axis

Major axis

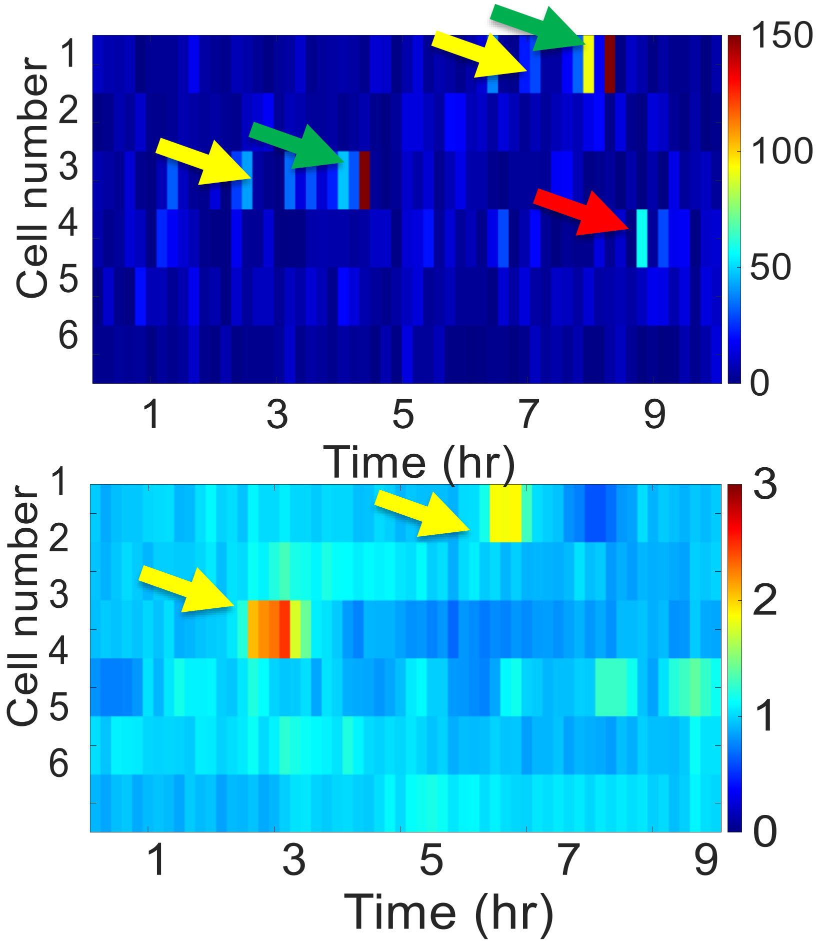
$$C_i^k = \frac{(P_i^k)^2}{4\pi(A_i^k)}$$

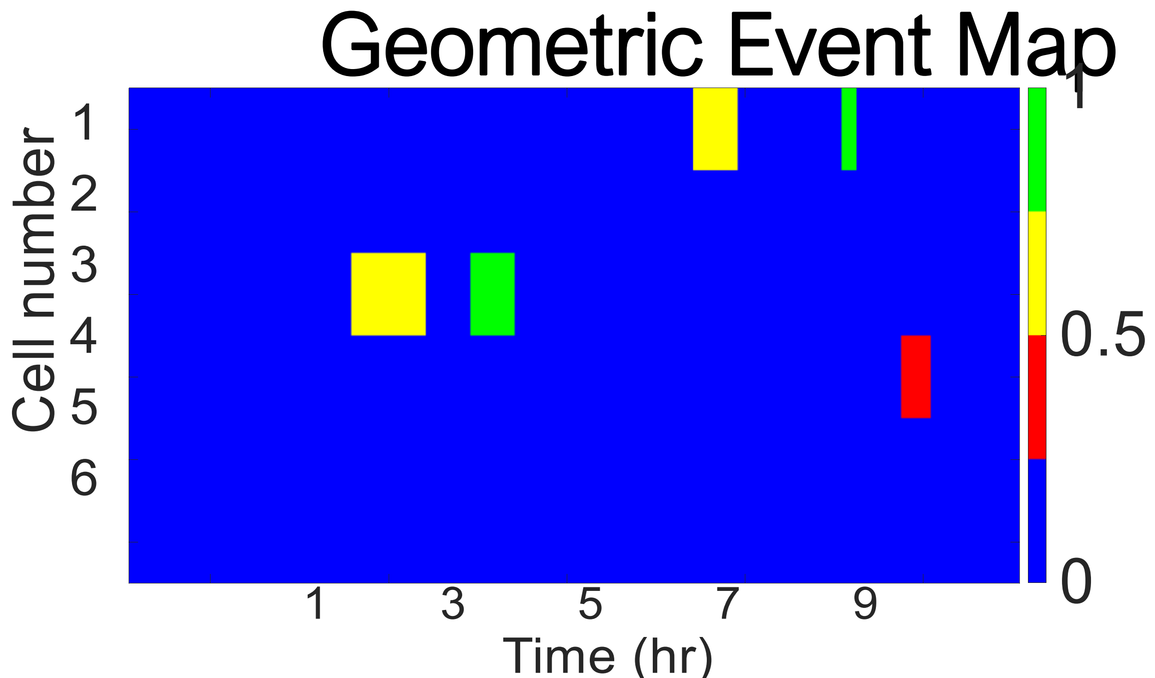
🟨 Circularity

🟩 Elongation

dArea/dt

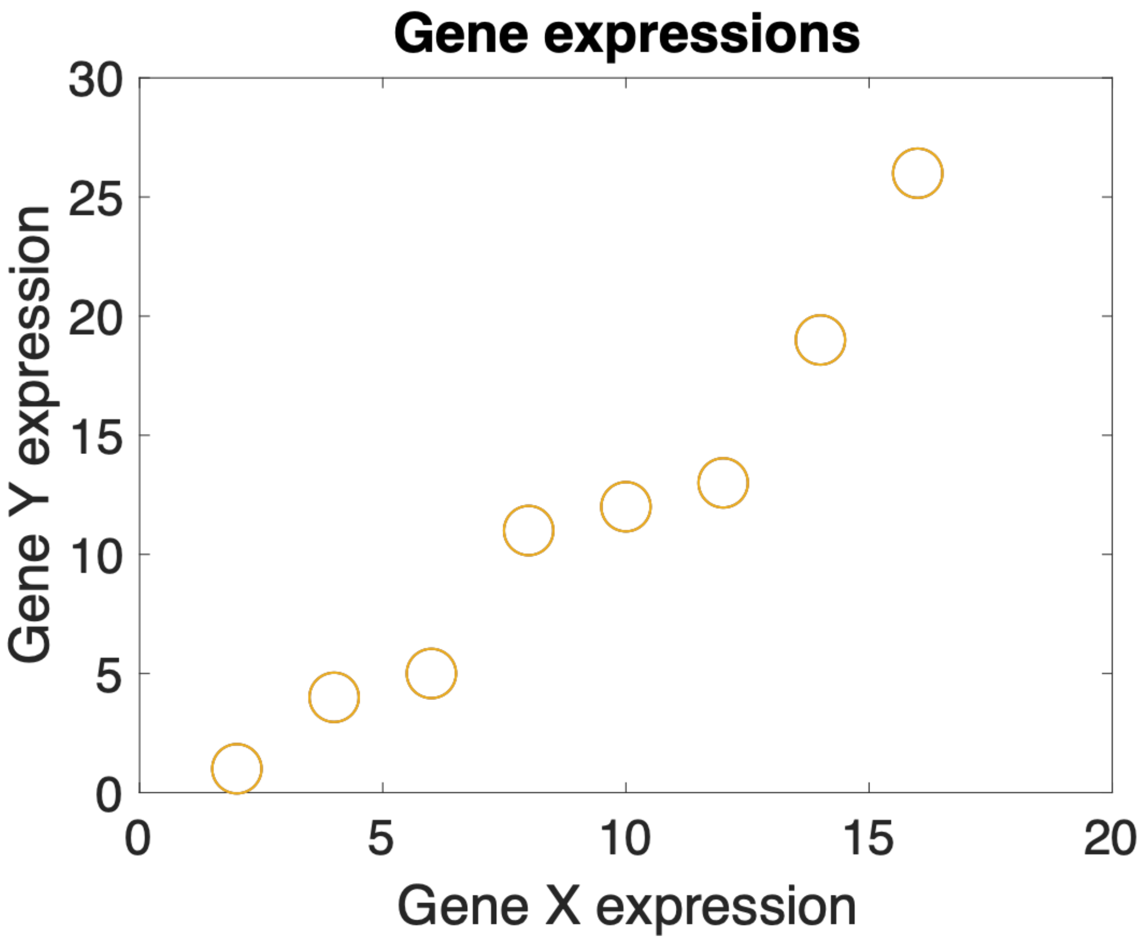
dIntensity/dt

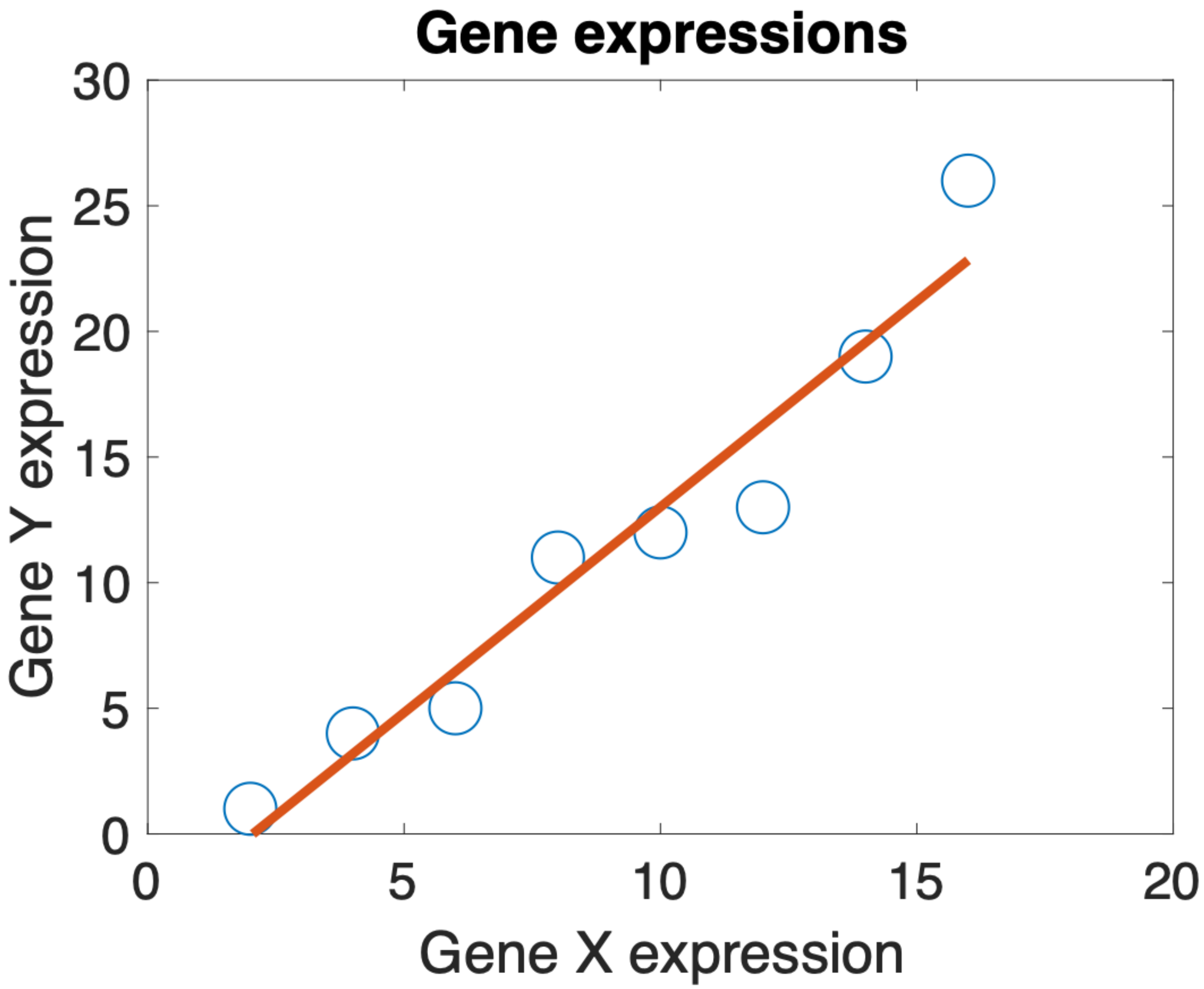




Linear Regression: Least Squares Fit

Fitting data y best in a least-square method.





[u i]=polyfit(a,b,1)

```
       u =
Slope    Intercept

1.6369   -3.3571
```

y=1.6369x-3.3571

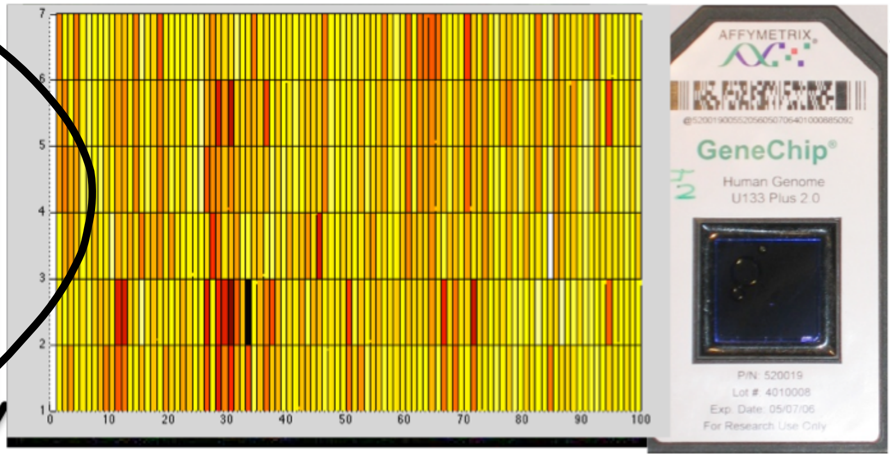
Which of the genes are expressed at the same time in cancer vs normal cells?

What are the genes that are expressed in cancer cells but not normal cells?

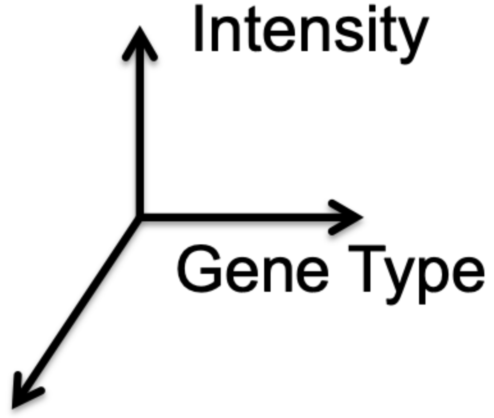
How the expression changes over time?

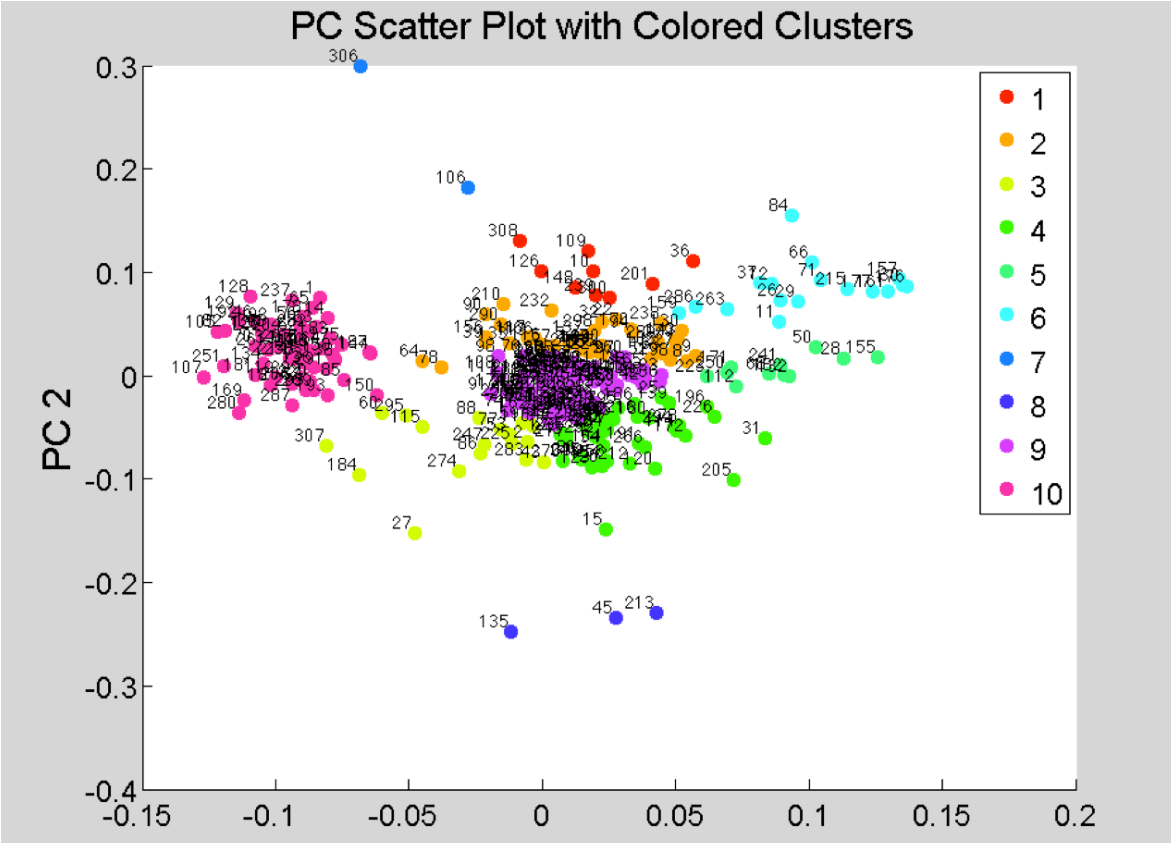
Type of gene

Time or tissue type

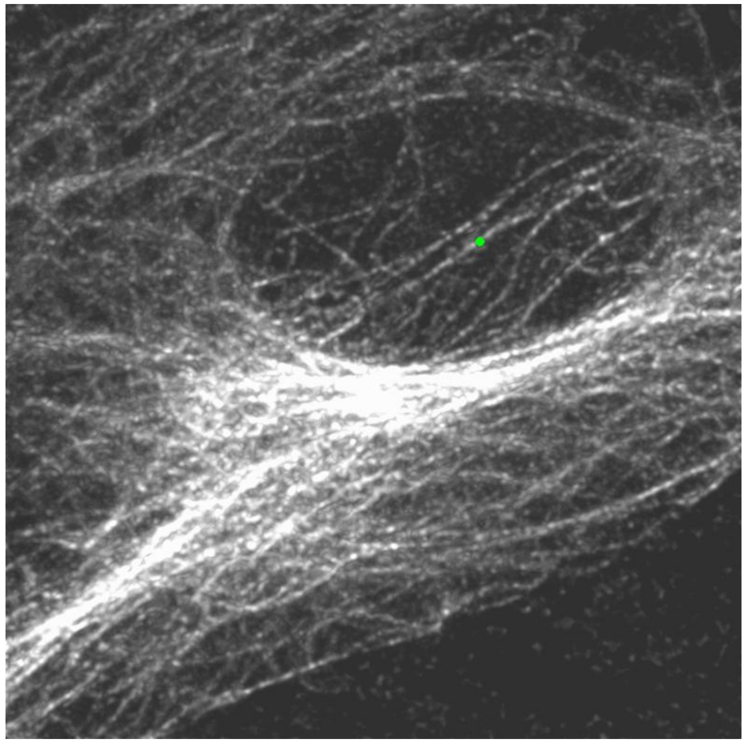


Colors define the intensity values

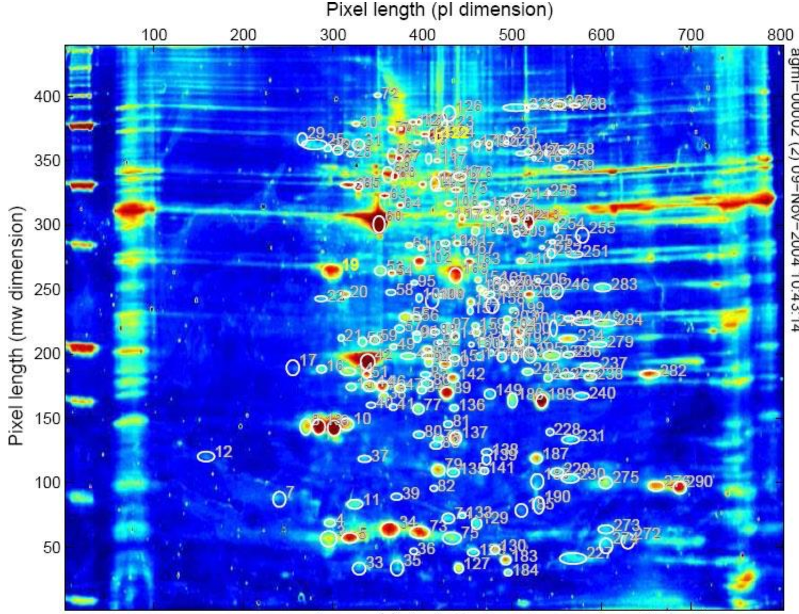




Genetics

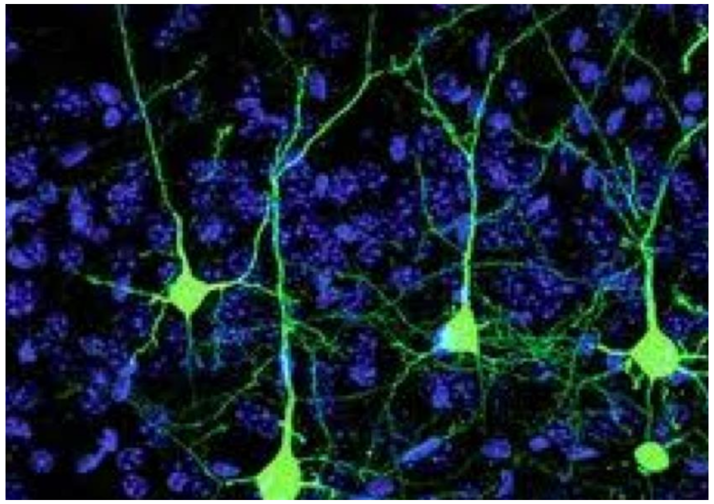


Molecular Biology



http://bioinformatics.musc.edu/Web2DE

Bioinformatics



Neuroscience

Computing is essential to solve many of these research problems

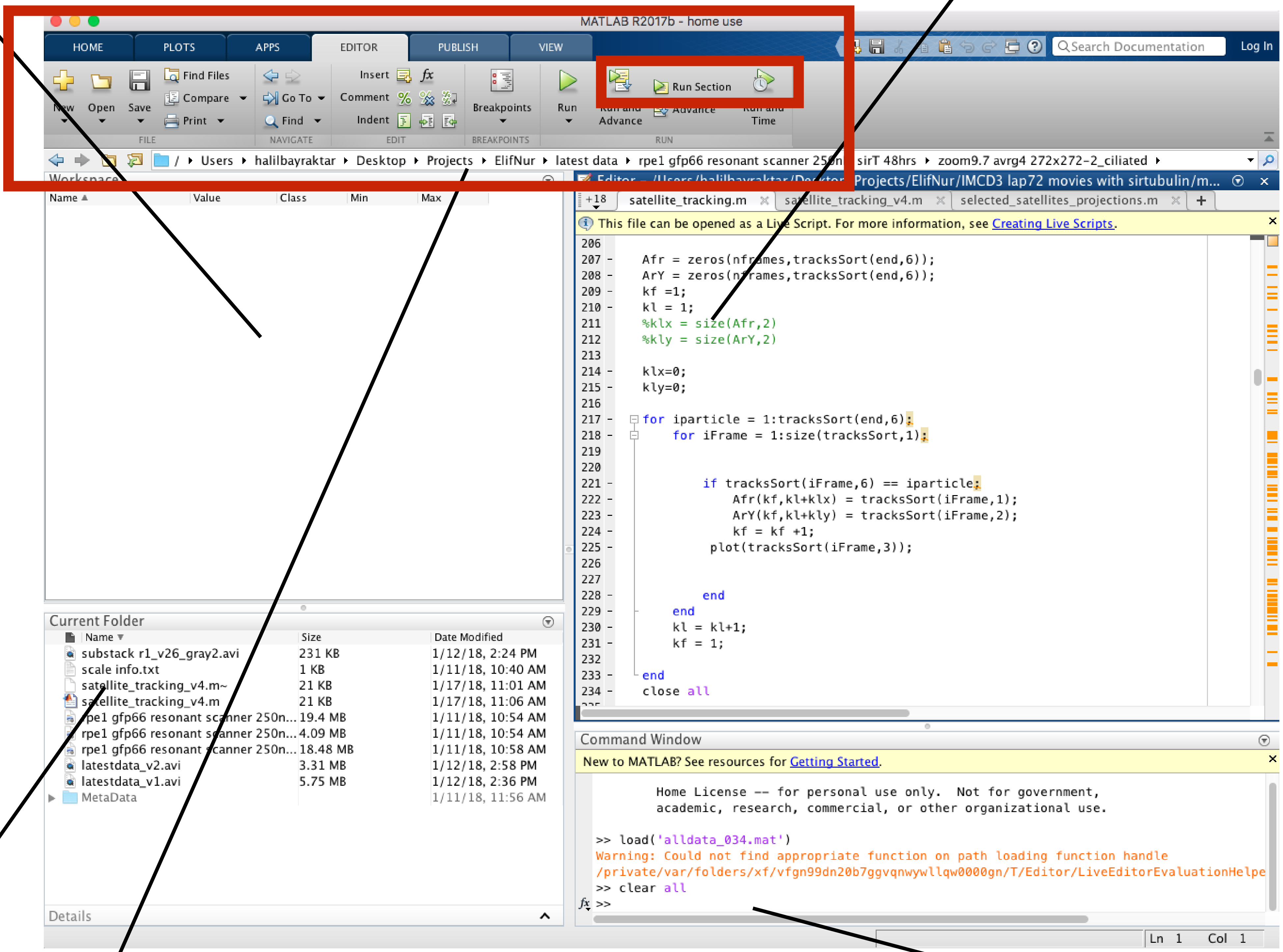
Artificial Intelligence

MATLAB USER INTERFACE

Workspace
to show all variables

Menu and other options

Coding



This file can be opened as a Live Script. For more information, see Creating Live Scripts.

```
    Afr = zeros(nframes,tracksSort(end,6));
    ArY = zeros(nframes,tracksSort(end,6));
    kf =1;
    kl = 1;
    %klx = size(Afr,2)
    %kly = size(ArY,2)

    klx=0;
    kly=0;

    for iparticle = 1:tracksSort(end,6);
        for iFrame = 1:size(tracksSort,1);


            if tracksSort(iFrame,6) == iparticle;
                Afr(kf,kl+klx) = tracksSort(iFrame,1);
                ArY(kf,kl+kly) = tracksSort(iFrame,2);
                kf = kf +1;
             plot(tracksSort(iFrame,3));


            end
        end
        kl = kl+1;
        kf = 1;

    end
    close all
```

Name	Size	Date Modified
substack r1_v26_gray2.avi	231 KB	1/12/18, 2:24 PM
scale info.txt	1 KB	1/11/18, 10:40 AM
satellite_tracking_v4.m~	21 KB	1/17/18, 11:01 AM
satellite_tracking_v4.m	21 KB	1/17/18, 11:06 AM
rpe1 gfp66 resonant scanner 250n...	19.4 MB	1/11/18, 10:54 AM
rpe1 gfp66 resonant scanner 250n...	4.09 MB	1/11/18, 10:54 AM
rpe1 gfp66 resonant scanner 250n...	18.48 MB	1/11/18, 10:58 AM
latestdata_v2.avi	3.31 MB	1/12/18, 2:58 PM
latestdata_v1.avi	5.75 MB	1/12/18, 2:36 PM
MetaData		1/11/18, 11:56 AM

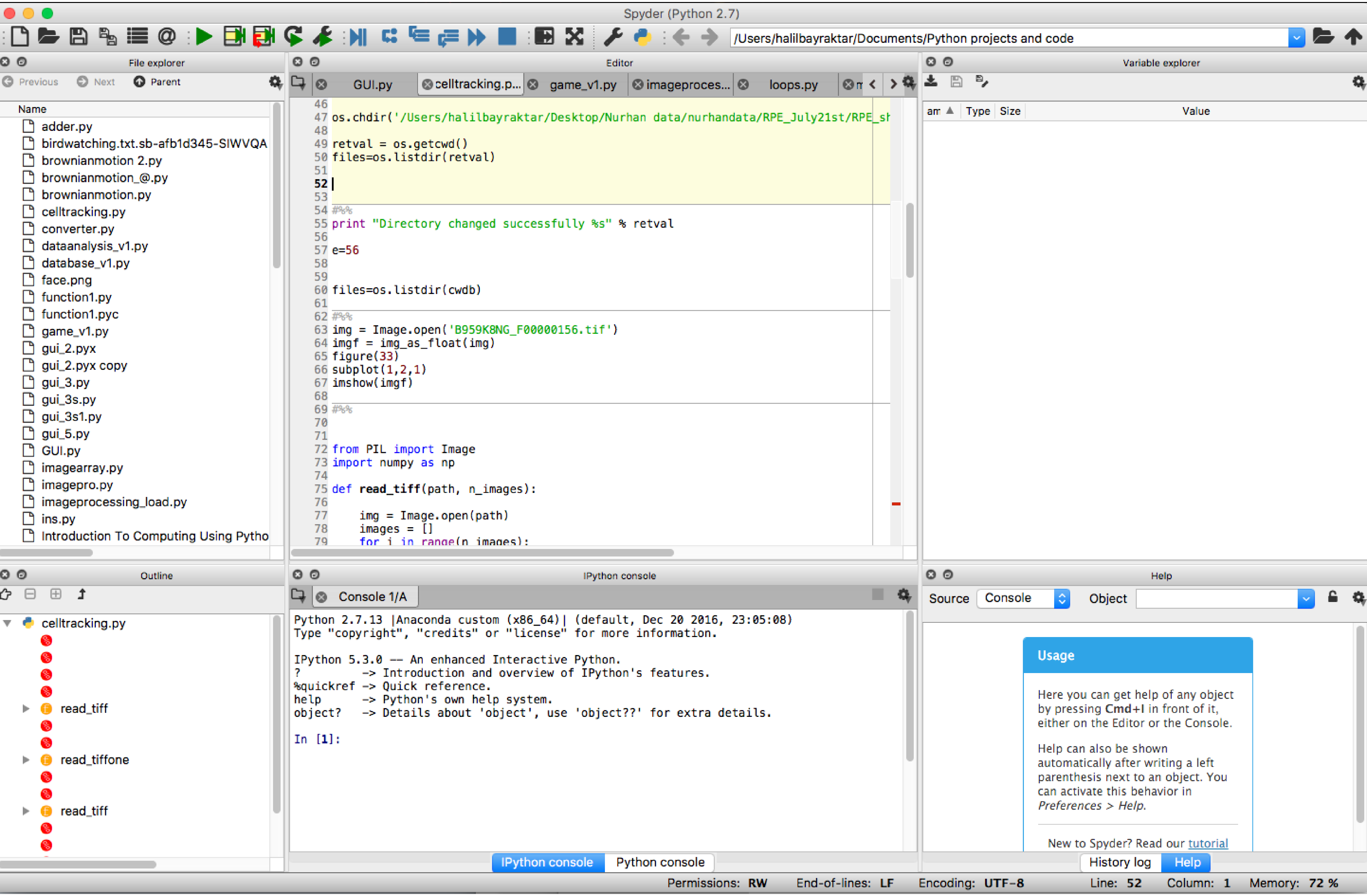
New to MATLAB? See resources for Getting Started.

```
          Home License -- for personal use only.  Not for government,
          academic, research, commercial, or other organizational use.

>> load('alldata_034.mat')
Warning: Could not find appropriate function on path loading function handle
/private/var/folders/xf/vfgn99dn20b7ggvqnwywllqw0000gn/T/Editor/LiveEditorEvaluationHelpe
>> clear all
>>
```

All files in a selected folder

Type commands here

Python Layout (open source platform)



adder.py
birdwatching.txt.sb-afb1d345-SIWVQA
brownianmotion 2.py
brownianmotion_@.py
brownianmotion.py
celltracking.py
converter.py
dataanalysis_v1.py
database_v1.py
face.png
function1.py
function1.pyc
game_v1.py
gui_2.pyx
gui_2.pyx copy
gui_3.py
gui_3s.py
gui_3s1.py
gui_5.py
GUI.py
imagearray.py
imagepro.py
imageprocessing_load.py
ins.py
Introduction To Computing Using Pytho

```
os.chdir('/Users/halilbayraktar/Desktop/Nurhan data/nurhandata/RPE_July21st/RPE_sh

retval = os.getcwd()
files=os.listdir(retval)



#%%
print "Directory changed successfully %s" % retval

e=56


files=os.listdir(cwdb)

#%%
img = Image.open('B959K8NG_F00000156.tif')
imgf = img_as_float(img)
figure(33)
subplot(1,2,1)
imshow(imgf)

#%%


from PIL import Image
import numpy as np

def read_tiff(path, n_images):

    img = Image.open(path)
    images = []
    for i in range(n_images):
```

celltracking.py
read_tiff
read_tiffone
read_tiff

```
Python 2.7.13 |Anaconda custom (x86_64)| (default, Dec 20 2016, 23:05:08)
Type "copyright", "credits" or "license" for more information.

IPython 5.3.0 -- An enhanced Interactive Python.
?         -> Introduction and overview of IPython's features.
%quickref -> Quick reference.
help      -> Python's own help system.
object?   -> Details about 'object', use 'object??' for extra details.

In [1]:
```

Usage

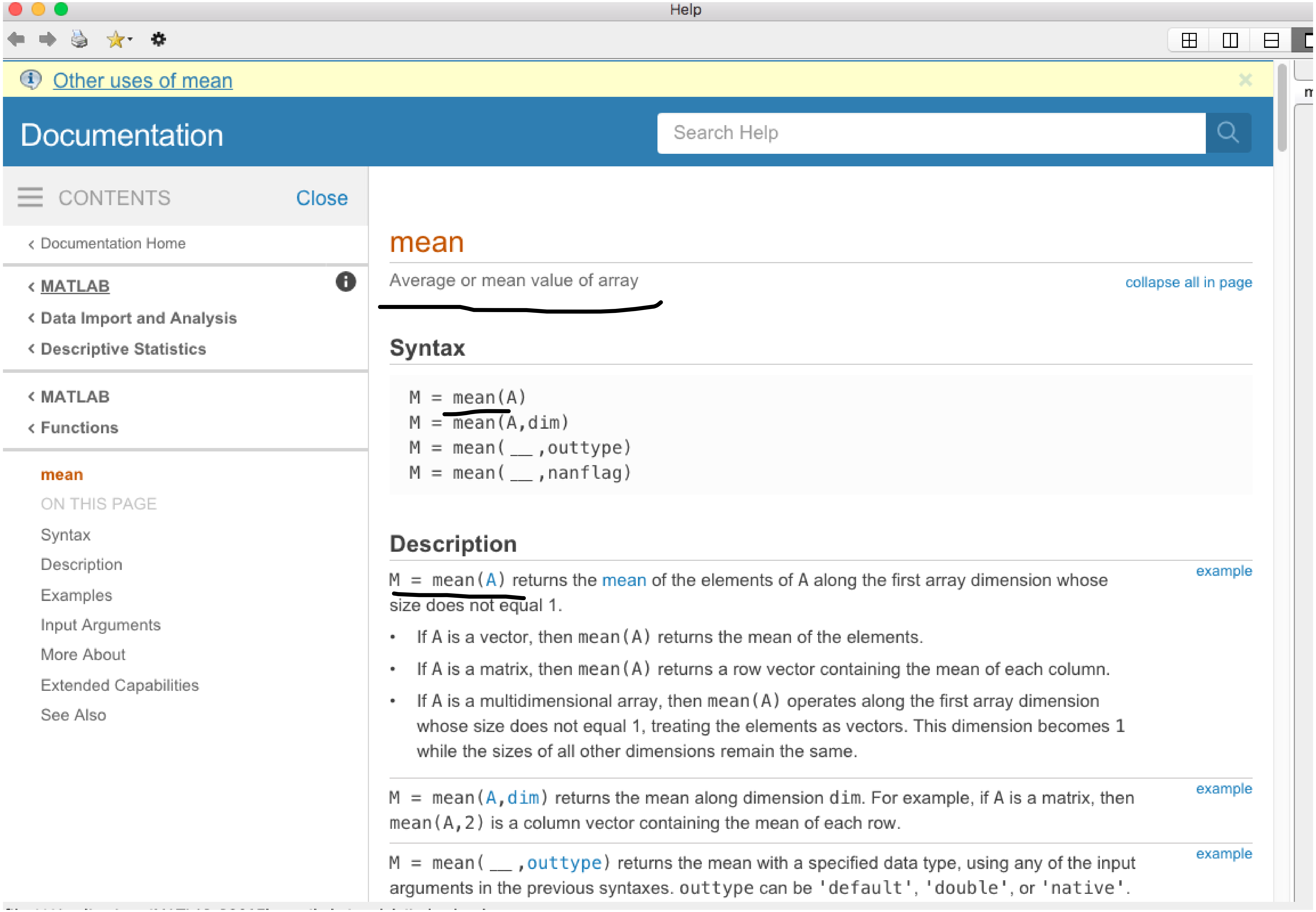
Here you can get help of any object by pressing **Cmd+I** in front of it, either on the Editor or the Console.

Help can also be shown automatically after writing a left parenthesis next to an object. You can activate this behavior in *Preferences > Help.*

New to Spyder? Read our tutorial

Permissions: RW   End-of-lines: LF   Encoding: UTF-8   Line: 52   Column: 1   Memory: 72 %

Matlab help



mean

Average or mean value of array

Syntax

```
M = mean(A)
M = mean(A,dim)
M = mean(___,outtype)
M = mean(___,nanflag)
```

Description

`M = mean(A)` returns the mean of the elements of A along the first array dimension whose size does not equal 1.

- If A is a vector, then `mean(A)` returns the mean of the elements.
- If A is a matrix, then `mean(A)` returns a row vector containing the mean of each column.
- If A is a multidimensional array, then `mean(A)` operates along the first array dimension whose size does not equal 1, treating the elements as vectors. This dimension becomes 1 while the sizes of all other dimensions remain the same.

`M = mean(A,dim)` returns the mean along dimension `dim`. For example, if A is a matrix, then `mean(A,2)` is a column vector containing the mean of each row.

`M = mean(___,outtype)` returns the mean with a specified data type, using any of the input arguments in the previous syntaxes. `outtype` can be `'default'`, `'double'`, or `'native'`.

Why MATLAB?

A single platform to load, sort, analyze, plot and save your data.

Beginner level syntax structure

Easy coding

Highly preferred in academia

Many open resources is available for help

Many functions are included

.mat and m. files

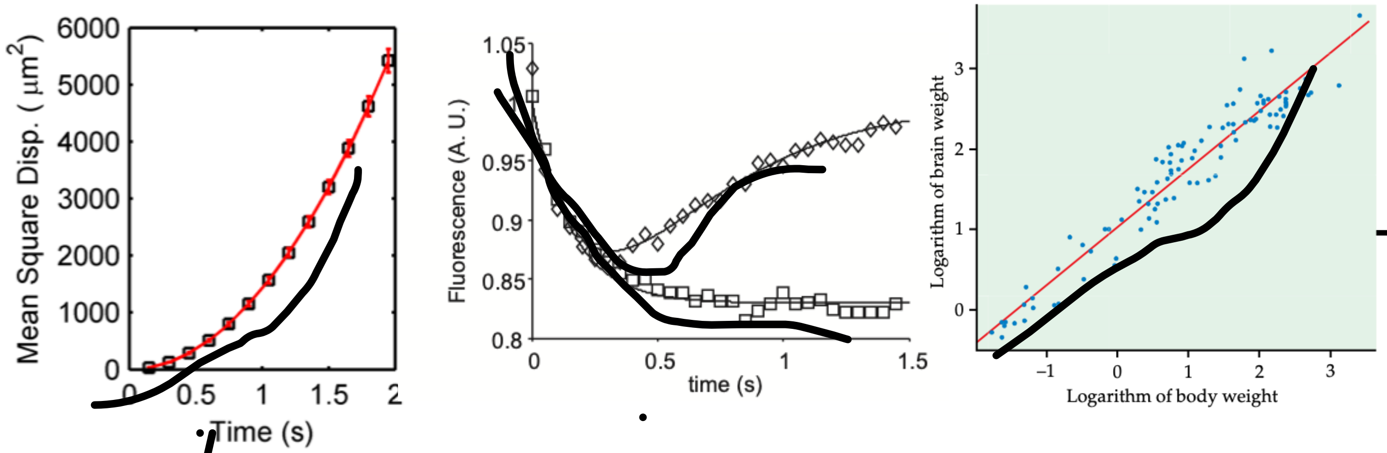
.m : Files that contain the computer code. There are two kinds of .m files. Scripts and function. Script files do not need any input value to execute, however the function requires one or more arguments to return the output

.mat files: files that contains the data and variable names

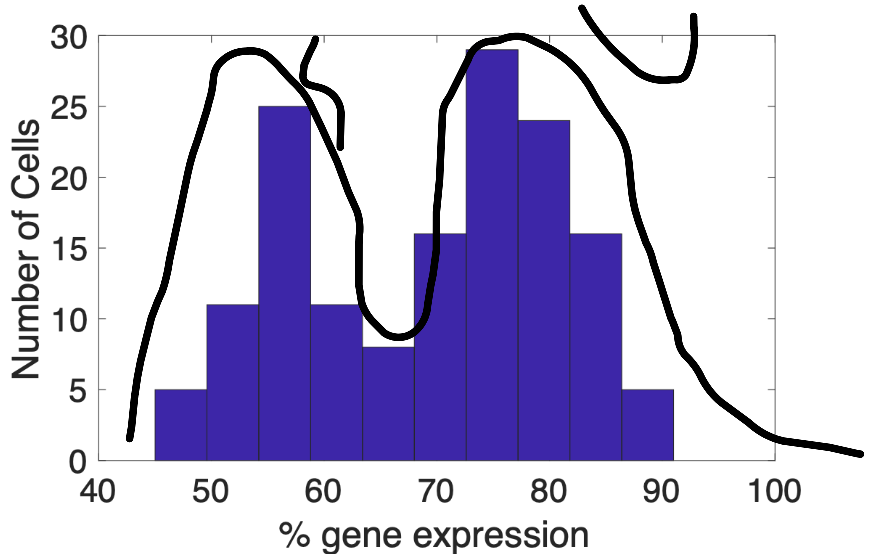
Writing code in computer vs playing notes in piano

You need practice, practice and practice to advance in writing codes.

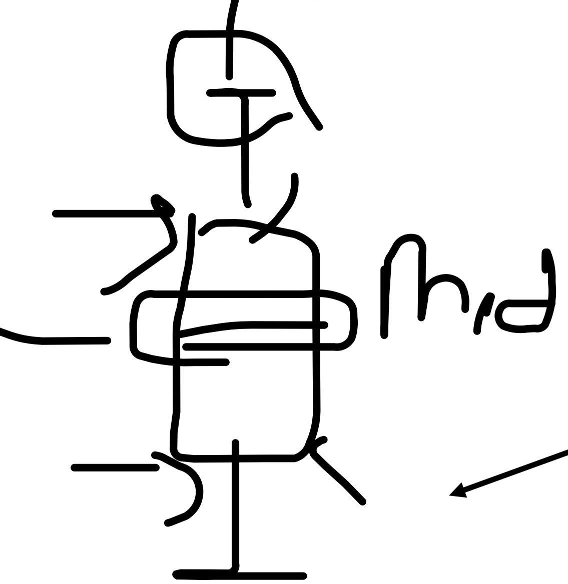
Description and Visualization of Statistical Data

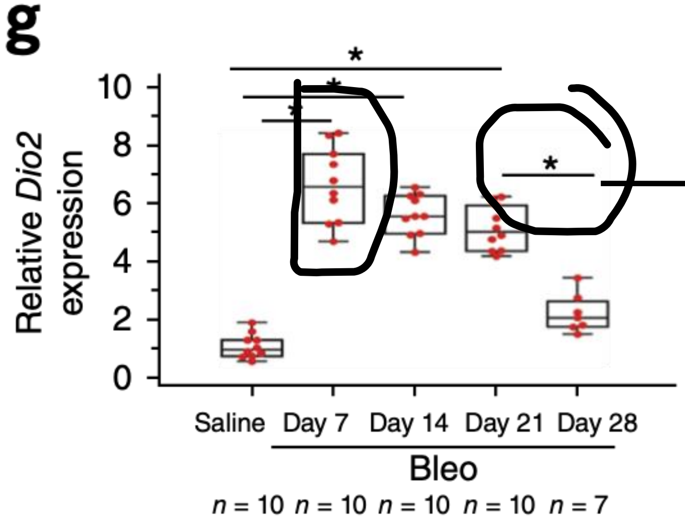


Linear Plot

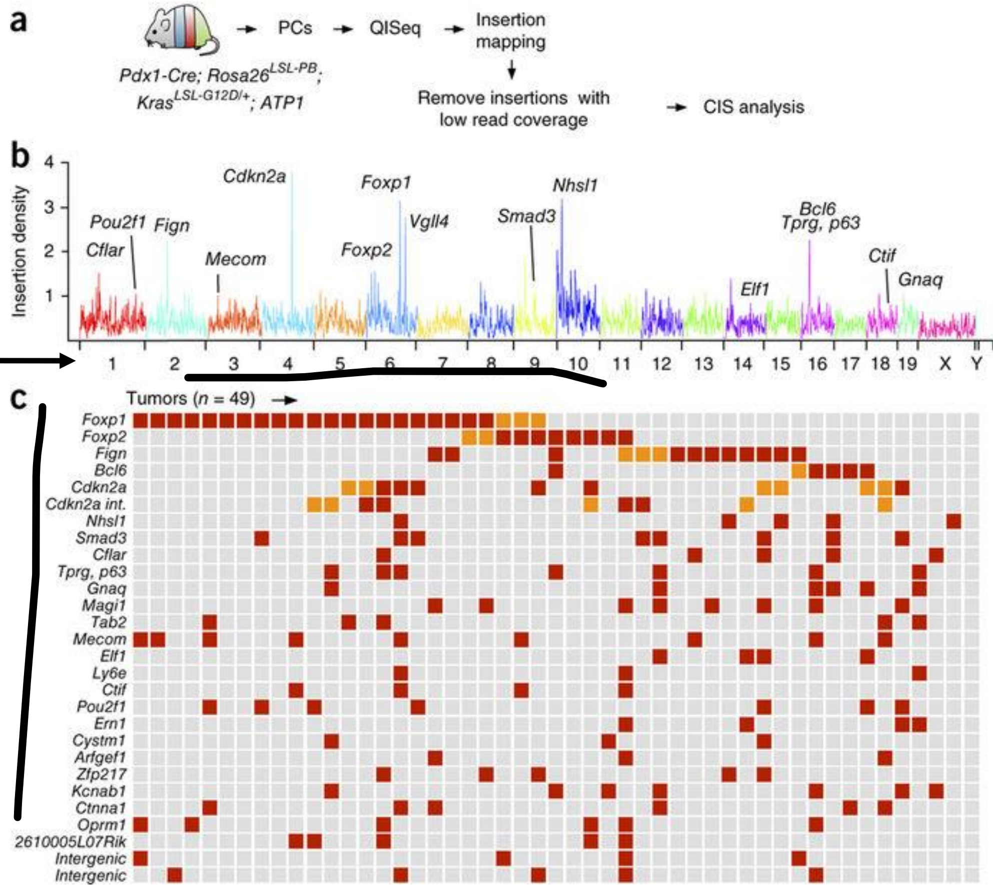


Histogram and bar plots



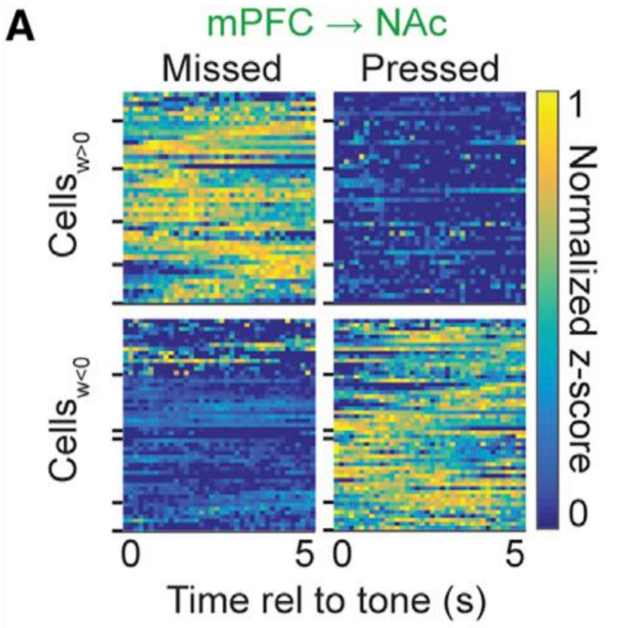


Box plot



3D plot

4D plot



Introduction to Matlab

MATLAB=Matrix laboratory developed by MathWorks Inc

High level language for computing, data visualization, modeling and programming

Matrices (1D, 2D etc) are the main components

NAME OF PUNCTUATION MARKS USED in MATLAB

Square brackets [ ]

hyphen -

curly brackets {}

Brackets, parentheses ()

Semi colon ;

Colon :

Comma ,

Dot .

Apostrophe ‘ or single quote

Double quote “”

Single quote ‘’

Backslash \

Slash /

Create a variable

Create a number variable

```
x=5;
y=10;
z= x+y;
```

```
x =
   2
y =
   5
z =
   7
```

Create a char variable

```
x = ‘Matlab’
```

You should use

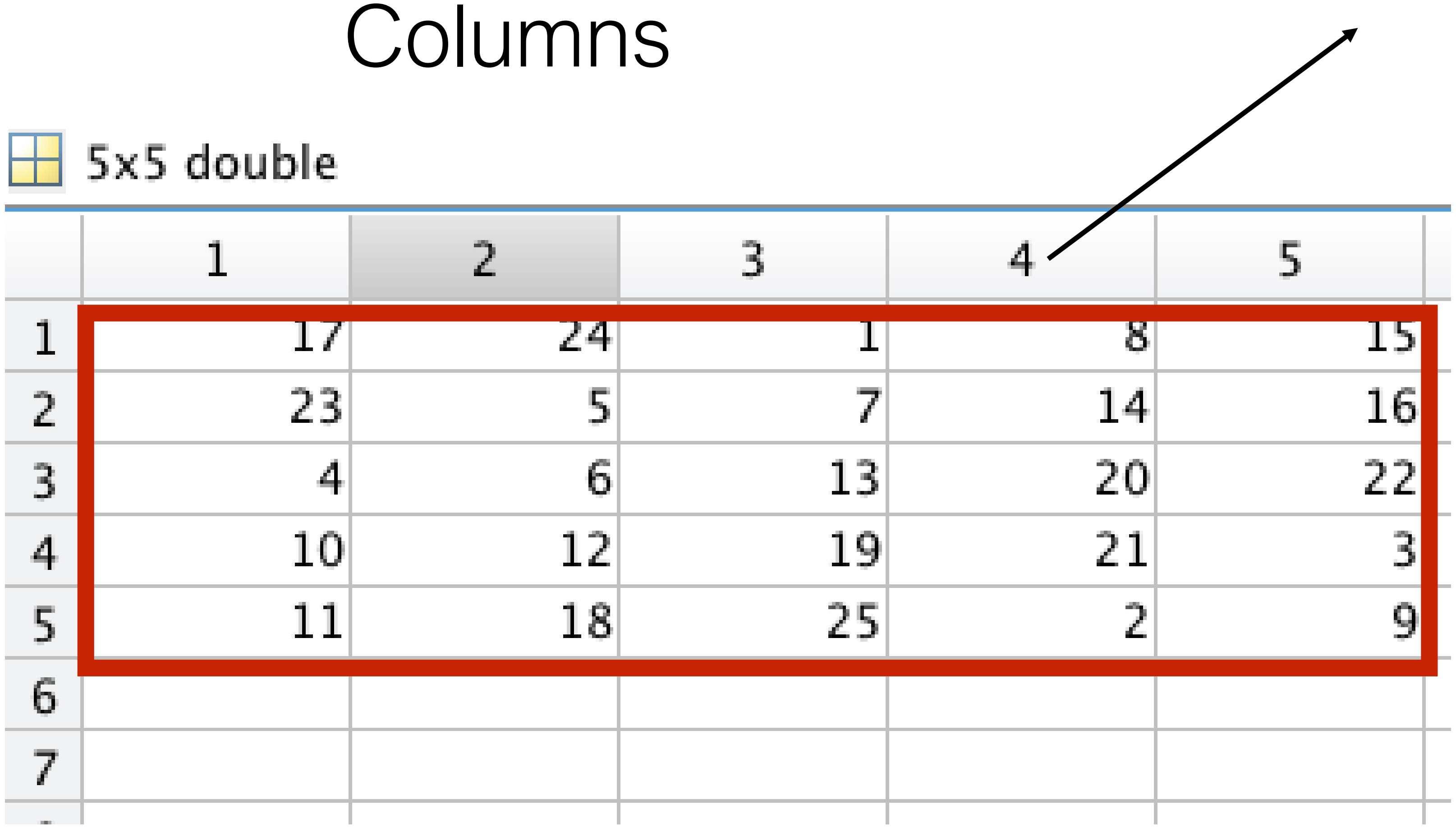
```
x=[1 2 3 4]
y=[1:2:10]
```

Index number (indices)

Columns

Rows

Data values



5x5 double

	1	2	3	4	5
1	17	24	1	8	15
2	23	5	7	14	16
3	4	6	13	20	22
4	10	12	19	21	3
5	11	18	25	2	9
6					
7					

5 by 5 matrix, 5x5

Create a vector

Generating a row and column vector

```
rowvec=[5,6,7,8,12]

colvec=[3;4;7;9;14]
```

```
rowvec =

     5     6     7     8    12


colvec =

     3
     4
     7
     9
    14
```

vector slicing

```
a=0:2:20
b=a(1,1:5)
c=a(1,5:end)
```

```
a =
    0    2    4    6    8   10   12   14   16   18   20
b =
    0    2    4    6    8
c =
    8   10   12   14   16   18   20
```

Create a vector

```
    x= 0:2:50
```

x = 0,2,4,….,100

```
size(x)
```

```
Ans
row and column
1          51
```

Indexing into a matrix is a means of selecting a subset of elements fro

Finding an element in row or column vector

```
% First entry is row index number
% Second entry is column index number

frel= rowvec(1,5)

fcel=colvec(3,1)


 rowvec=[5,6,7,8,12]

 colvec=[3;4;7;9;14]
```

```
 frel =

    12

>> fcel

 fcel =

    7
```

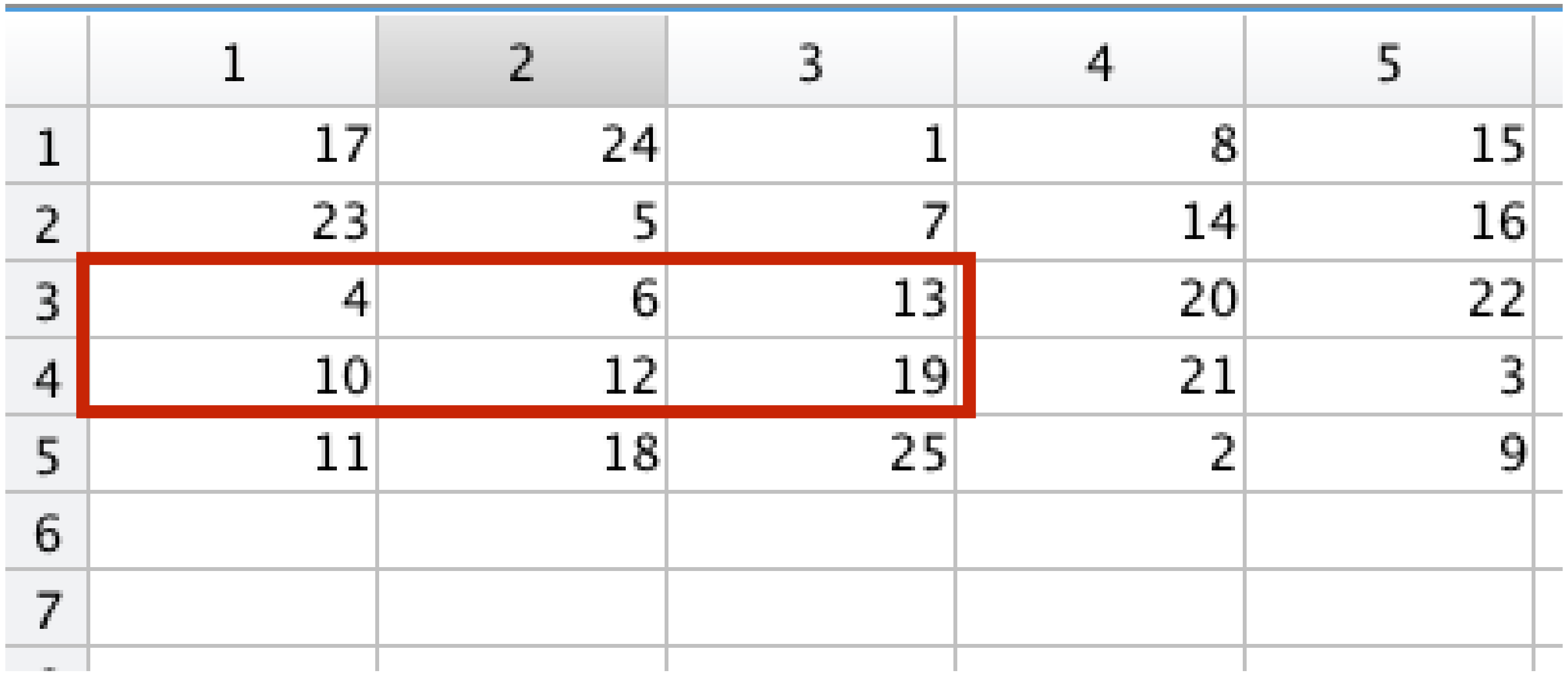
Operators

Sum
Division
Multiplication

Select a region from matrix

Column1 Column2

row 1
row 2



	1	2	3	4	5
1	17	24	1	8	15
2	23	5	7	14	16
3	4	6	13	20	22
4	10	12	19	21	3
5	11	18	25	2	9
6					
7					

Row Column

```
b=a([3,4],[1,2,3])
```

```
b =

 4    6    13
10   12    19
```

Finding and selecting elements in a matrix

```
z=magic(4)
z([1,2],[3,4])
z(1,1)
z(2)
```

```
z =

    16     2     3    13
     5    11    10     8
     9     7     6    12
     4    14    15     1
```

```
ans =

     3    13
    10     8
```

```
ans =

    16
```

```
ans =

     5
```

Some Common Scalar Mathematical Operations

	Operation	M-file	
Subtraction	X-y	X-y	
Addition	x + y xx+y+y	x + y	
Multiplication	xy	x*y	
Division	x/y	x/y	
Power	x^y	x^Y	
Exp	e^x	exp(x)	
Log	log10(x)	log10(x)	
Nat Log	ln(x)	log(x)	
	cos(X) 0	Cos(x)	
	Sin(x)LSEP9x)	Sin(x)	
	$x^{1/2}$	sqrt(x)	