

Introducing Chemical Engineers to Coding, Data and Visualization through MATLAB

Jason E. Bara

University of Alabama

Chemical & Biological Engineering

AIChE Annual Meeting

Orlando, FL

November 12, 2019

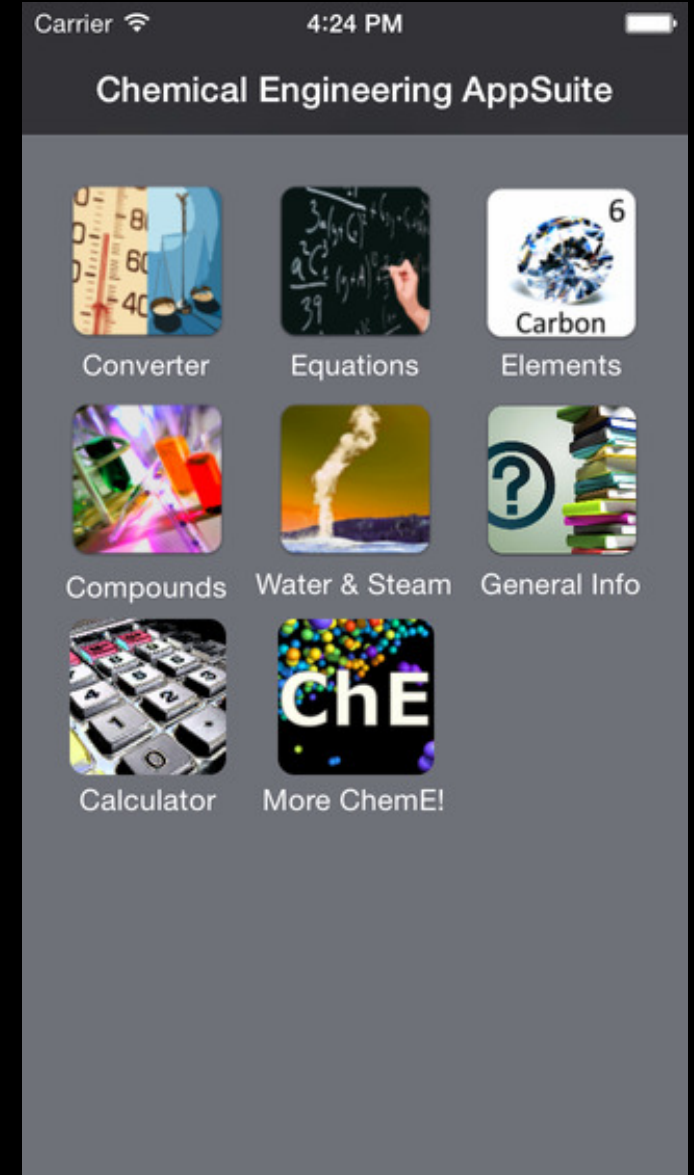
Collaborators

- Dr. Vincent F. Scalfani [@vfscalfani](#)
 - PhD in Chemistry, Colorado State University
 - Associate Professor, UA Libraries
 - Science & Engineering Librarian
- Ali Al Alshaikh
 - Current MS Student (& future PhD student)
 - BS, Chemical Engineering, UA 2017



The ChemE App

- Since January 2012
 - iPhone and iPad (iOS)
- Properties for 1000+ compounds
 - Leveraged into different modules
- Equations of State
 - PR, RK, vdW, SRK, compressibility
- Calculators
 - Antoine Equation, Heat Capacity, Pressure Head, etc.
- Delivery of reference material via PDF
- Unit Converter
 - Integrated to automatically solve equations in any units



Example iPhone Screenshots

Verizon 11:45 AM

[Back](#) methanol

methanol
Compound

methyl alcohol
Common Names

CH₄O
Formula

67-56-1
CAS Number

[View in Wikipedia](#)

32.042
Molecular Weight

0.791
Density (g/mL) at 20 °C

-97.7
Melting Temperature (°C)

64.6
Boiling Temperature (°C)

35.254
Heat of Vaporization (kJ/mol)

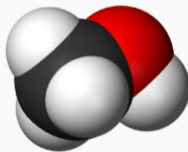
Verizon 11:45 AM

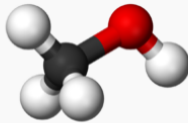
[Back](#)

Methanol

Not to be confused with [menthol](#) or [methanal](#).

Methanol

$$\begin{array}{c} \text{H} \\ | \\ \text{H}-\text{C}-\text{OH} \\ | \\ \text{H} \end{array}$$


$$\begin{array}{c} \text{H} & \text{H} \\ & \diagdown \quad / \\ & \text{C}-\text{O}-\text{H} \\ & / \\ \text{H} \end{array}$$


Verizon 11:46 AM

[Back](#) Antoine Equation

Antoine

octane

P:

T:

GO

$\log p^*(\text{mm Hg}) = A - B/(T(^{\circ}\text{C})+C)$

A: 6.9094

B: 1349.82

C: 209.385

Lower Temperature Limit: 19°C

More iPhone Screenshots

Verizon 11:45 AM

[Back](#) Pressure

atm

mm Hg

bar

mbar

Torr

psi

psf

Pa

kPa

MPa

GPa

Verizon 11:49 AM

[Back](#) Steam

☒ Saturated ☐ Superheated

T:

P:

GO Clear

Liquid Cp: 4.182 kJ/kg-K

Vapor Cp: 1.912 kJ/kg-K

Liquid Cv: 4.139 kJ/kg-K

Vapor Cv: 1.44 kJ/kg-K

Liquid Density: 997 kg/m³

Vapor Density: 0.02288 kg/m³

Liquid Enthalpy: 104.8 kJ/kg

Verizon 11:48 AM

[Back](#) Reynolds Number

D:

v:

ρ :

μ :

Re: Unitless

GO CLEAR

Equation $Re = (Dv\rho)/\mu$

Even More iPhone Screenshots

Carrier 4:35 PM

< Back Peng Robinson

P: 22 atm

Vbar: 1.9695 L/mol

T: 333 °C

GO CLEAR

Z: 0.8712

a: 10.39 L²·bar/mol²

b: 0.04143 L/mol

Acentric Factor ω : 0.556

κ : 1.149

Critical Temperature: 513 K

Carrier 4:34 PM

CMD Vars Exit

t=6.934 (t redefined)
s=t^2 (s defined)
s = 48.08036
x=s+t/2 (x defined)
x = 51.54736
sin(x) = 0.958551
mean(s, t, x) = 35.52057
stddev(s, t, x, 2) = 22.77345
4 + 4% = 4.16

2*sin($\pi/2$)

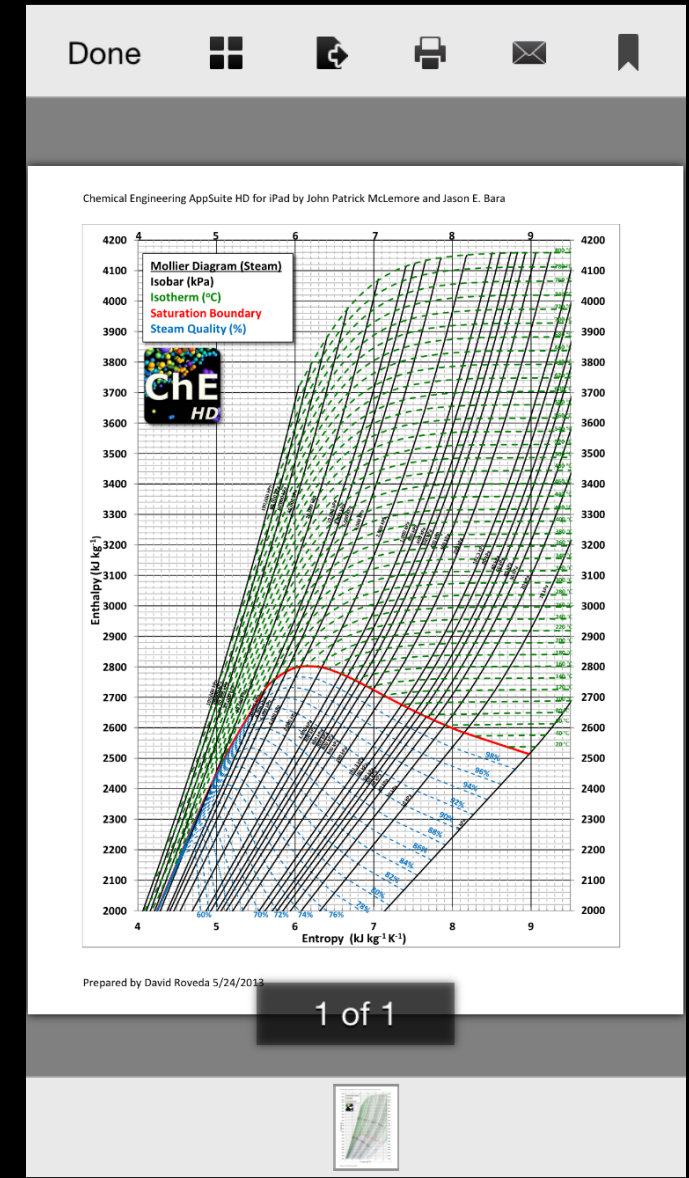
+ - * / ^ % ! > < π =

1 2 3 4 5 6 7 8 9 0

- / : ; () \$ & @ "

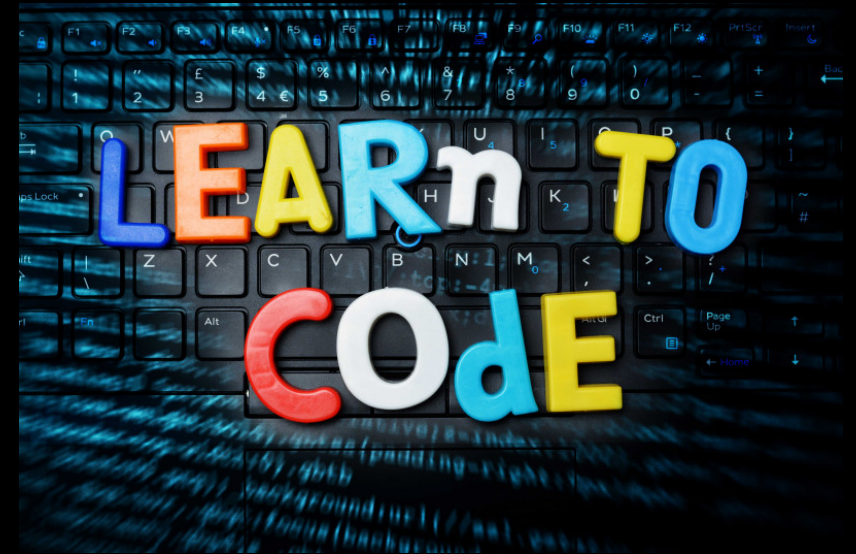
#+= . , ? ! ' $\times$

ABC space return



“Coding” Approach to Teaching ChemE

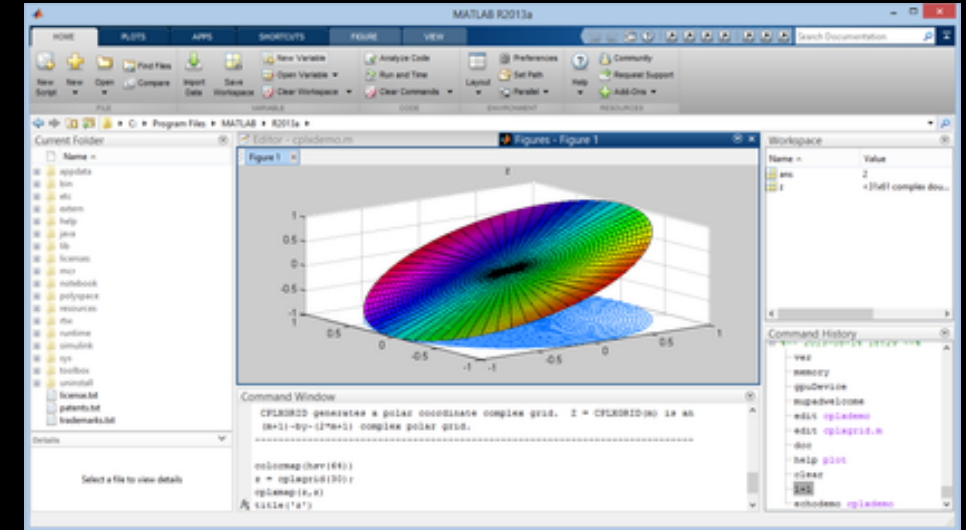
- Most of our upperclassmen have never written code
 - Developed elective course for seniors and graduate students
- Solve fun, interesting and relatable problems
 - Not necessarily “traditional” chemical engineering
 - Require students to provide a new problem on every assignment
 - Analytical solutions NOT accepted as final answers
- “Discover” statistics and probability
 - Not a required course in undergraduate curriculum
 - Random number generation and many iterations
- Collect freely available data from online resources
- Projects, reports and oral presentations
 - Communicate results and methods!



<https://robogarden.ca/blog/leverage-of-coding>

Why MATLAB? (vs. Open Source)

- UA owns site license
 - Students *already* paid for access through fees
 - Mac and PC both supported
- Visualization
 - Can be as simple as `plot(x)`
 - Matlab graphs now much easier to format via GUI
- Toolboxes
 - Lots of useful and cool features already built in
- Well-documented, no longer as intimidating
 - Mathworks provides excellent resources online
 - Forums provide Q&A for many less obvious coding concepts
 - The answer is usually only a Google search away...



Example Questions & Solutions

Two People with Same Birthday

In a class of 50 students and 1 professor (i.e. this class), what is the probability that at least two people share the same birthday?

- **Key concepts/constructs**

- Random number (1-365)
- Array
- For loop
- If statement
- Plot
- Commenting/documenting



1980



1960

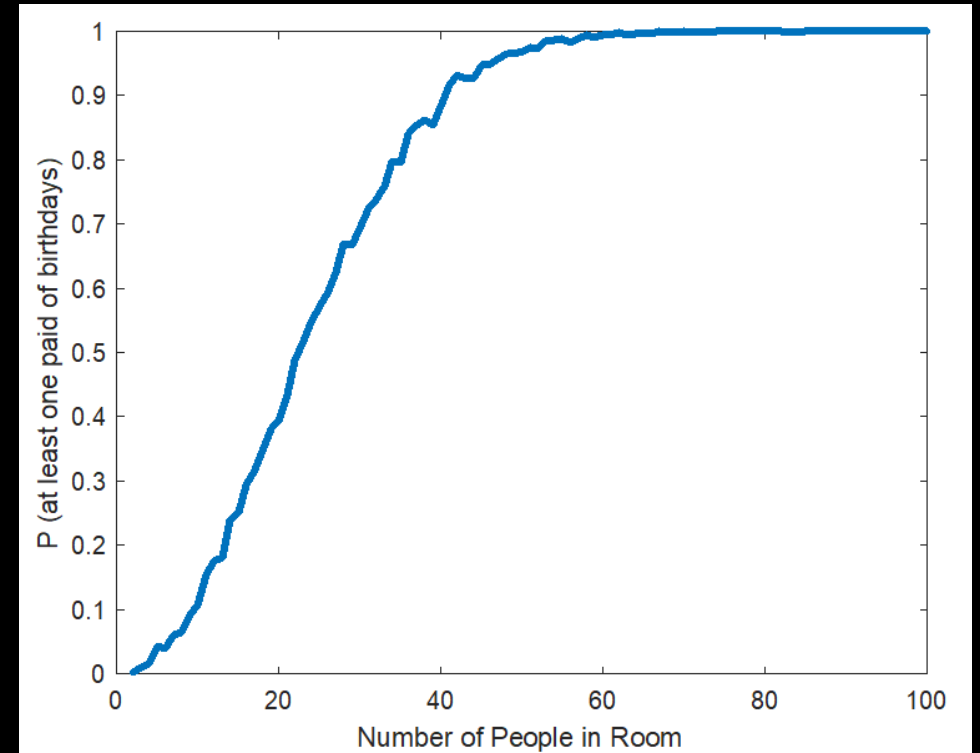
September 10

```

%Probability of at least 1 pair of b-days vs. # of people in room
%room size (i) will be increased from 2 people to 100 people in increment of 1
eachmatch = []; %array that will hold fraction of matches for each room size
totalruns = 10000;
for i = 2:1:100
    matches = 0;
    for j = 1:1:totalruns
        %Generate a random array of i numbers between 1 and 365
        bdays = randi(365, i, 1);
        %How many unique random numbers were generated?
        check = unique(bdays);
        %Is the number of unique elements less than i?
        if numel(check) < i %If true, there must be at least one match
            matches = matches + 1;
        end
    end
    eachmatch = [eachmatch, matches/totalruns];
end
xaxis = [2:1:100]; %Create x-axis against which to plot eachmatch
plot(xaxis, eachmatch)

```

The line becomes smoother with more runs!



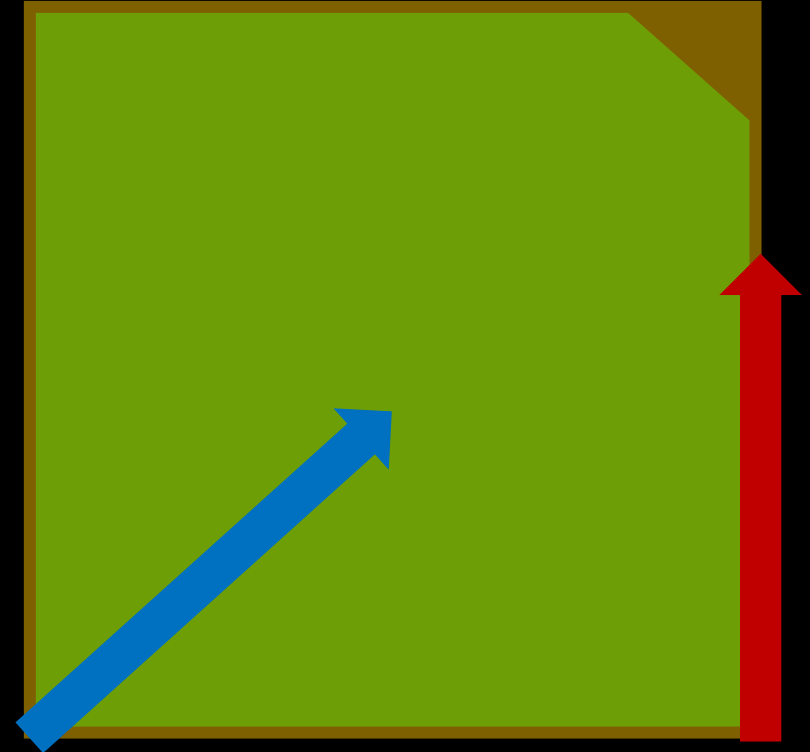
N = 51: > 97% probability

Farmer vs. Ram

A ram and farmer start at two corners of a pen $(0, 0)$ and $(0, 1)$ respectively. The farmer runs to the gate $(1, 1)$ to escape the charging ram. The ram runs directly at the farmer at all times. What is the *minimum* speed that the ram needs to run relative to the farmer to catch him before he can exit?

- **Key concepts/constructs**

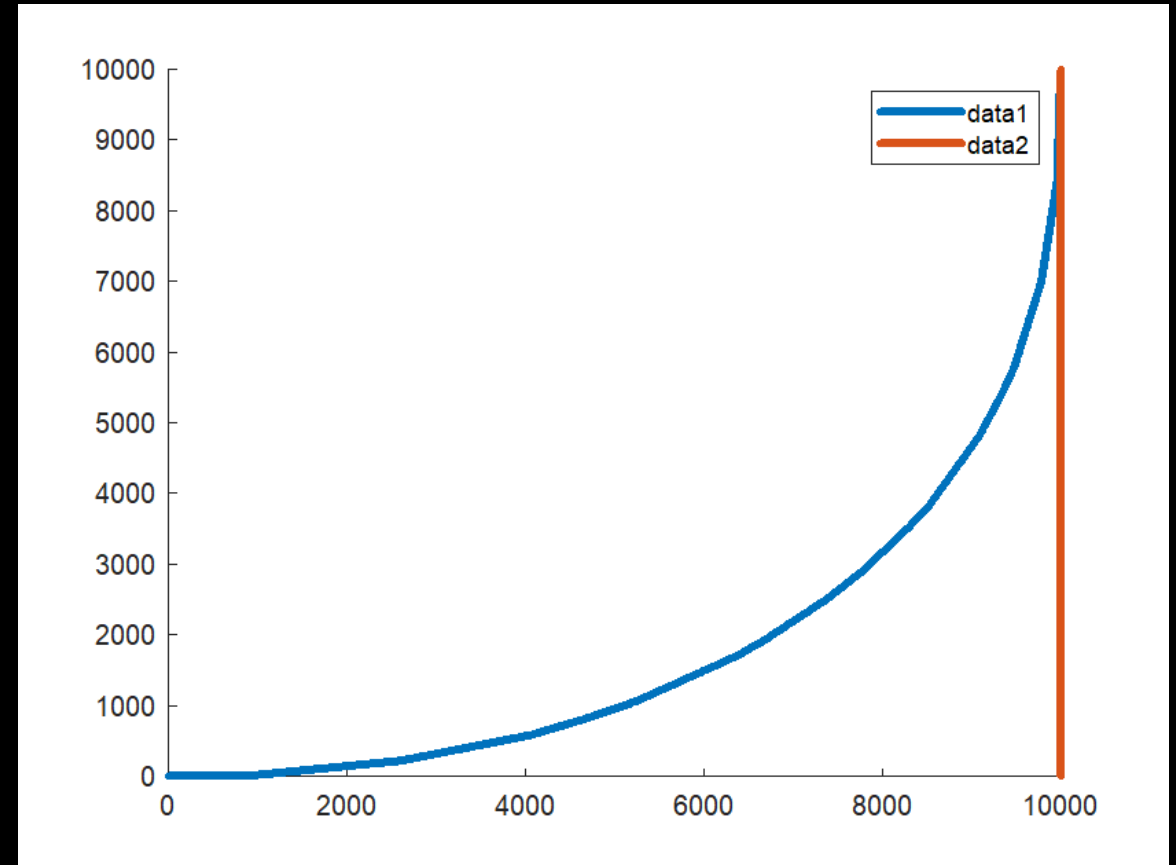
- Optimization/search
- Mesh/grid



```

%How fast does the ram need to run to catch the farmer?
%The grid is the field (x by y)
gridsize = 10000;
%set initial x, y coordinates for farmer and ram
farmer = [gridsize,0];
ram = [0, 0];
%The location of the gate... it doesn't move
gate = [gridsize, gridsize];
fspeed = 5; %farmer moves this many grid spaces with each move)
moves = gridsize/fspeed;
a = 1.61 %how much faster does ram need move than farmer to catch him/her
rspeed = a*fspeed;
farmerx = farmer(1);
farmery = farmer(2);
ramx = ram(1);
ramy = ram(2);
for i = 1:1:moves
    %farmer moves segment
    farmer = [gridsize, i*fspeed];
    farmerx = [farmerx, farmer(1)];
    farmery = [farmery, farmer(2)];
    %calculate movement of ram by comparing coordinates
    direction = farmer - ram;
    directionScaled = direction/norm(direction);
    %below we are rounding to the nearest integer (hence 0)
    ram = round((ram + rspeed.*directionScaled), 0);
    ramx = [ramx, ram(1)];
    ramy = [ramy, ram(2)];
    if ram(1) == gridsize
        break
    end
end
end

```



Ram will catch farmer if ~1.61 times as fast...

$$\mathbf{v}_{\text{ram}} = \Phi \cdot \mathbf{v}_{\text{farmer}}$$

Number Chain

Using the integers from 1 to 100, inclusive, and organize them into a chain. Each number can only be used once and each number must be adjacent in the chain to one of its factors or multiples. For example, you might build the chain:

4, 12, 24, 6, 60, 30, 10, 100, 25, 5, 1, 97

In this case, you have no numbers left to place after 97, leaving you with a finished chain of length 12. What is the longest chain you can build? What strategies are most useful for building longer chains? Is there a good “seed” number which with to start the chain?

- **Key concepts/constructs**

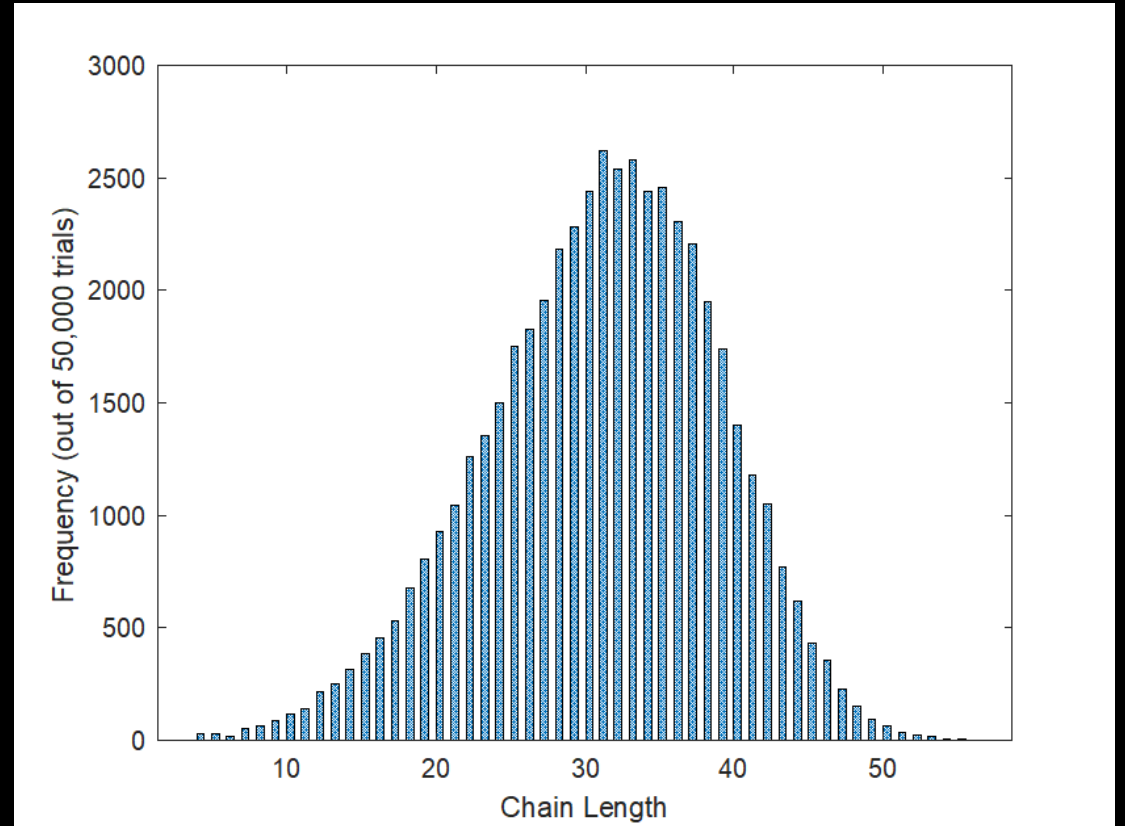
- While loop
- Histogram
- Machine learning

1	2	3	4	5	6	7	8	9	10
11	12	13	14	15	16	17	18	19	20
21	22	23	24	25	26	27	28	29	30
31	32	33	34	35	36	37	38	39	40
41	42	43	44	45	46	47	48	49	50
51	52	53	54	55	56	57	58	59	60
61	62	63	64	65	66	67	68	69	70
71	72	73	74	75	76	77	78	79	80
81	82	83	84	85	86	87	88	89	90
91	92	93	94	95	96	97	98	99	100

```

%Number Chain Builder
chainlengths = [];
chains3 = [];
%Initialize Array Outside of Loop to improve code execution time
maximum = 100;
startarray = 1:1:maximum;
%remove all primes greater than 53 because they're all effectively the same
badguys = [59, 61, 67, 71, 73, 79, 83, 89, 97]; %we proved this via machine learning
startarray = setdiff(startarray, badguys);
chainstarts = [];
for i = 1:1:50000
    chainstate = "Alive";
    newtrial = startarray;
    activechain = [];
    %Don't allow to start with 1
    element = randi([2, 91]);
    number = newtrial(element);
    chainstarts = [chainstarts, number];
    activechain = [activechain, number];
    %Remove element from startarray
    newtrial(newtrial == number) = [];
    while chainstate == "Alive"
        %find multiples available
        multiples = newtrial(rem(newtrial, activechain(end))==0);
        factors = newtrial(rem(activechain(end), newtrial)==0);
        choices = [factors, multiples];
        choices = sort(choices);
        if numel(choices) > 1 && choices(1) == 1
            choices(1) = [];
        end
        if isempty(choices)
            chainstate = "Dead";
            chainlengths = [chainlengths, length(activechain)];
            if length(activechain) == 3
                chains3 = [chains3; activechain];
            end
            break
        else
            arraysize = size(choices);
            element = randi(arraysize);
            number = choices(element);
            activechain = [activechain, number];
            %Remove element from startarray
            newtrial(newtrial == number) = [];
        end
    end
end
end
histogram(chainlengths)

```



Chain length of 56 achieved with limited constraints in just 50,000 iterations

- Don't start with 1
- Remove primes of 59 or greater

Proven maximum is 77

Tracking Movement

- A frog starts at a random position on a 10 x 10 map and then “jumps” to a random adjacent space (up, down, diagonal) with each move. The frog cannot jump off the map. On average, how many moves does it take before the frog has landed on every space? Use an animated heatmap to visualize the frog’s most efficient run.
- **Key concepts/constructs**
 - Heatmap
 - Animation



```
%Frog Jumping on Pond
```

```
x = 10; %grid dimension
```

```
y = 10; %grid dimension
```

```
totals = [];
```

```
for trial = 1:1:1
```

```
    grid = zeros(x, y);
```

```
    frogx = randi([1, x]);
```

```
    frogy = randi([1, y]);
```

```
    %frog starts at this random spot, increment each spot by 1 every time it touches a spot
```

```
    grid(frogx, frogy) = grid(frogx, frogy) + 1;
```

```
    frog = heatmap(grid)
```

```
    while min(grid, [], 'all') == 0
```

```
        %calculate valid moves
```

```
        moves = [-1, -1, -1, 0, 0, 1, 1, 1; -1, 0, 1, -1, 1, -1, 0, 1];
```

```
        validmoves = [];
```

```
        for i = 1:1:8
```

```
            xmove = moves(1, i);
```

```
            ymove = moves(2, i);
```

```
            testx = frogx + xmove;
```

```
            testy = frogy + ymove;
```

```
            if testx >= 1 && testx <= x && testy >= 1 && testy <= y
```

```
                validmoves = [validmoves; xmove, ymove];
```

```
            end
```

```
        end
```

```
        %How many valid moves do we have and let's randomly pick one
```

```
        [a, b] = size(validmoves);
```

```
        randmove = randi([1, a]);
```

```
        xmove = validmoves(randmove, 1);
```

```
        ymove = validmoves(randmove, 2);
```

```
        %move the frog!
```

```
        frogx = frogx + xmove;
```

```
        frogy = frogy + ymove;
```

```
        %Update the grid & animate heatmap
```

```
        grid(frogx, frogy) = grid(frogx, frogy) + 1;
```

```
        frog = heatmap(grid);
```

```
        pause(0.01);
```

```
    end
```

```
    totals = [totals; sum(grid, 'all')];
```

```
end
```

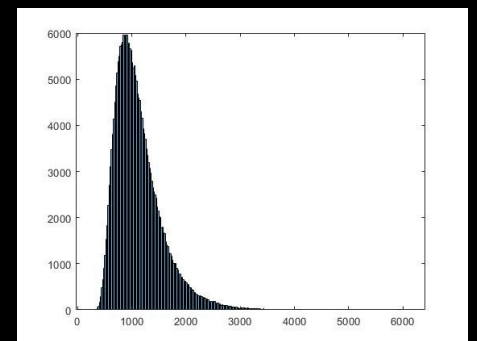
1	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	1
6	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0
	1	2	3	4	5	6	7	8	9	10

Results from 250,000 trials:

Min hops = 270

Max hops = 6119

Average = 1127



Text Analytics & HTML Parsing

- Built-in Wordcloud
 - Example: Romeo & Juliet
- HTML parsing
 - Automate retrieval of data/metadata from websites
 - Retrieve names of universities with only a list of URLs
 - Retrieve chemical information from repositories (e.g., cactus.nih.gov)

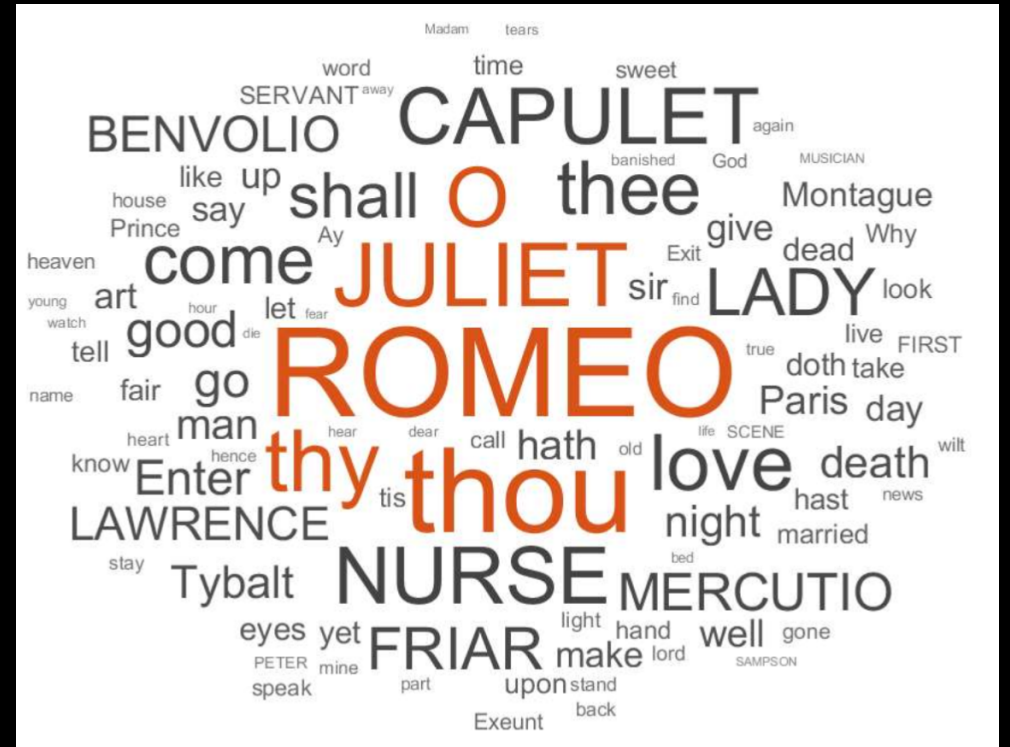


Image Analysis

- Image analysis
 - Images are just a 3-D matrix
 - RGB: 0-255 for each x,y pixel
- “Average” color of the house in this thermal image?
- % of house surface area is hotter than average?



Course Outcomes & Observations

- Several students that have underperformed in traditional ChemE classes have greatly outperformed expectations in this course
- Among undergrads (N = 43), the Top 4 highest averages in course are females!
 - 10 out of Top 12 also females
- Provide chemical engineering students opportunities to be truly creative and some amazing things can happen

Independent Projects

Independent Projects

- Engage undergraduate students in semester (or longer) projects using MATLAB, big data and other tools
 - Can yield publications
- Chemical identifiers
 - PubChem
- Literature analysis
 - SciFinder structure search
 - Quantification and mapping

Programmatic Compilation of Chemical Data and Literature from PubChem using MATLAB

Vincent F. Scalfani, Serena C. Ralph, Ali Al Alshaikh & Jason E. Bara
Chemical Engineering Education (In Review)

```
Search PubMed Collection

% Search pubchem pubmed collection for
% references containing meshheadings "simvastatin" and meshsubheadings "chemical synthesis"
% (associated with CID_query)

litQpubmedF = [sdq '{"select":"**","collection":"pubmed","where":{"ands":{"cid":"' CID_query '" ...
    "meshheadings":"simvastatin", "meshsubheadings":"chemical synthesis"}},"start":1,"limit":10}');
options = weboptions('Timeout', 60, 'ContentType','json');
litQpubmedF = webread(litQpubmedF, options);

% Display results: pmid, articletitle, articlejournalname, articleauth, pubmed link
for k = 1:length(litQpubmedF.SDQOutputSet.rows)
    pmid = litQpubmedF.SDQOutputSet.rows(k,1).pmid
    articletitle = litQpubmedF.SDQOutputSet.rows(k,1).articletitle
    articleauth = litQpubmedF.SDQOutputSet.rows(k,1).articleauth
    articlejournalname = litQpubmedF.SDQOutputSet.rows(k,1).articlejournalname
    citation = litQpubmedF.SDQOutputSet.rows(k,1).citation

    pubmed_root = 'https://www.ncbi.nlm.nih.gov/pubmed/?term=';
    pmid = num2str(pmid);
    pubmed_link = [pubmed_root pmid];
    fprintf('a href = " %s ">%s</a>\n', pubmed_link, pubmed_link)
    disp(' ')
end
```

```
Compile Bibliometric Data into a Table

% select only the numerical count data to export
SSq_bibtable = SSq_bibtable_validate(:, {'IsoSMI' 'CID' 'substance' 'patent' 'pubmed' ...
    'thiemechemistry' 'springernature' 'assay' 'bioactivity'})

SSq_bibtable = 58x9 table

   IsoSMI      CID      substance      patent      pubmed      thiemech...      springer...      assay      bioactivity
   _____      _____      _____      _____      _____      _____      _____      _____      _____
1  "C1C=C(C)C...  "3035817"      "42"      "17"      "690"      "0"      "645"      "504"      "523"
2  "C1C=C(C)C...  "490561"      "29"      "6"      "690"      "0"      "5"      "85"      "37"
3  "C1C=C(C)C...  "3813"      "27"      "8"      "689"      "0"      "0"      "47"      "52"
4  "C1C=C(C)C...  "9981348"      "21"      "2"      "2"      "0"      "5"      "2"      "14"
5  "CCCCC...      "3844"      "23"      "1"      "381"      "0"      "228"      "53"      "85"
6  "CCCCC...      "16219549"      "10"      "0"      "380"      "0"      "0"      "3"      "2"
7  "C1C=C(C)C...  "9912412"      "12"      "2"      "689"      "0"      "0"      "0"      "0"
8  "CCCCC...      "9850141"      "7"      "0"      "383"      "0"      "0"      "97"      "100"
9  "CCCCC...      "454202"      "34"      "22"      "414"      "0"      "5"      "2"      "2"
10 "C1C=C(C)C...  "11615648"      "10"      "0"      "901"      "0"      "0"      "0"      "0"
```

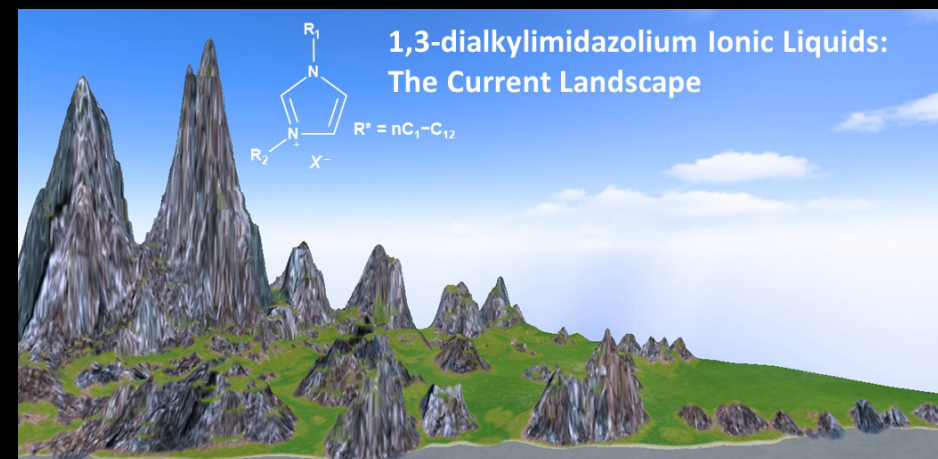
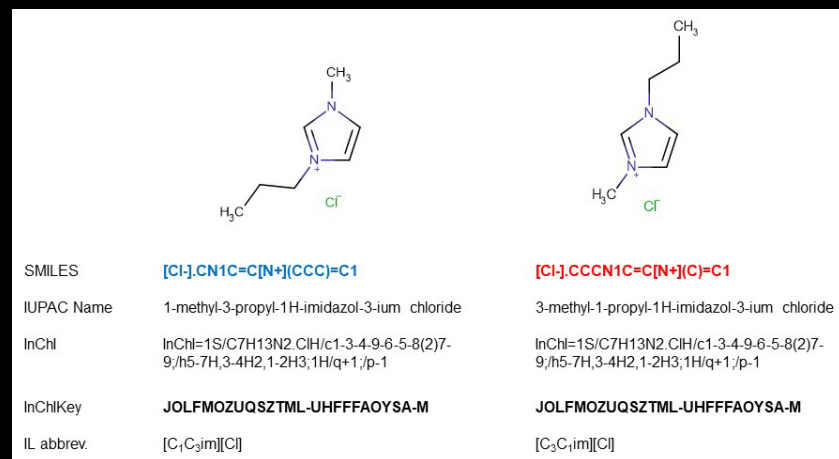
Repository:
github.com/vfscalfani/MATLAB-cheminformatics

Analysis of the Frequency and Diversity of 1,3-Dialkylimidazolium Ionic Liquids Appearing in the Literature

Vincent F. Scalfani, Ali Al Alshaikh & Jason E. Bara

Ind. Eng. Chem. Res. **2018**, *57*, 15971-15981

- Generate SMILES for library of cations and anions with MATLAB script



Thank You!

Jason E. Bara



jbara.eng.ua.edu



jbara@eng.ua.edu



[@ProfBara](https://twitter.com/ProfBara)