../Resources/references.bib

Python Quantitative Project

Ryan Greenup

August 10, 2020

Contents


```
code /home/ryan/Dropbox/Studies/QuantProject/Current/Python-Quant/ & disown
```

1 What the unit involves.

Writing and presenting.

2 Python

Two parts:

1. Figure out the python
2. IMPORTANT: MUST have new math
   (a) Dr. Hazrat may have some new math to use.

3 Matrix Exponentiation

3.0.1 Implementation in Sympy

The Matrix Exponential is implemented in areas of:

- Graph Centrality modelling [**?**]
- Systems of Linear Differential Equations [**?**, Ch. 8.4]
- Theory of Algebraic Lie Groups [**?**, Ch. 2]

However the method to implement matrix exponentiation provided by the documentation [**?**] and referenced in the development repository [**?**] does not appear to be implemented very well, for example the following provides a very long result:

```
from __future__ import division
from sympy import *
x, y, z, t = symbols('x y z t')
k, m, n = symbols('k m n', integer=True)
f, g, h = symbols('f g h', cls=Function)
init_printing()
init_printing(use_latex='mathjax', latex_mode='equation')


import pyperclip
def lx(expr):
    pyperclip.copy(latex(expr))
    print(expr)
```

```
A = Matrix([
    [11, 12, 13],
    [21, 22, 23],
    [31, 32, 33]
])

  expr = exp(A)
  expr.doit()
```

$$\begin{bmatrix} -\frac{1}{-\frac{33}{94}+\frac{5\sqrt{1149}}{94}}\left(\frac{-\frac{6552}{-\sqrt{1149}-22}+552}{\left(-\sqrt{1149}-22\right)\left(-\frac{59\sqrt{1149}}{95}-\frac{1837}{95}\right)}-\frac{26}{-\sqrt{1149}-22}\right)-\frac{1}{\frac{33\sqrt{1149}}{2303}+\frac{5745}{2303}}\left(-1-\frac{1}{-\frac{33}{94}+\frac{5\sqrt{1149}}{94}}\left(\frac{26}{-\sqrt{1149}-22}-\frac{-\frac{}{-\sqrt{}}}{(-\sqrt{1149}-22)}\right.\right. \\ -2-\frac{1}{\left(-\frac{1837}{95}+\frac{59\sqrt{1149}}{95}\right)\left(\frac{33\sqrt{1149}}{2303}+\frac{5745}{2303}\right)e^{-33+\sqrt{1149}}}\left(-1-\frac{1}{-\frac{33}{94}+\frac{5\sqrt{1149}}{94}}\left(\frac{26}{-\sqrt{1149}-22}-\frac{-\frac{6552}{-\sqrt{1149}-22}+552}{\left(-\sqrt{1149}-22\right)\left(-\frac{59\sqrt{1149}}{95}-\right.}\right.\right. \\ -\frac{1}{-\frac{33}{94}+\frac{5\sqrt{1149}}{94}}\left(\frac{-\frac{6552}{-\sqrt{1149}-22}+552}{\left(-\sqrt{1149}-22\right)\left(-\frac{59\sqrt{1149}}{95}-\frac{1837}{95}\right)}-\frac{26}{-\sqrt{1149}-22}\right)-\frac{1}{\frac{33\sqrt{1149}}{2303}+\frac{5745}{2303}} \end{bmatrix}$$

Simplifying this result doesn't seem to help either:

```
simplify(expr)
```

$$\begin{bmatrix} \frac{1}{12\left(-1065889+33298\sqrt{1149}\right)e^{\sqrt{1149}}}\left(-8625947e^{33+2\sqrt{1149}}-2131778e^{\sqrt{1149}}-2032943e^{33}+74651\sqrt{1149}e^{33}+66596\sqrt{1149}e^{\sqrt{1149}}+2\right. \\ \frac{1}{6\left(-1065889+33298\sqrt{1149}\right)e^{\sqrt{1149}}}\left(-66949e^{33+2\sqrt{1149}}-66596\sqrt{1149}e^{\sqrt{1149}}-2064829e^{33}+61128\sqrt{1149}e^{33}+2131778e^{\sqrt{1149}}+5\right. \\ \frac{1}{12\left(-1065889+33298\sqrt{1149}\right)e^{\sqrt{1149}}}\left(-236457\sqrt{1149}e^{33+2\sqrt{1149}}-6226373e^{33}-2131778e^{\sqrt{1149}}+66596\sqrt{1149}e^{\sqrt{1149}}+169861\sqrt{1149}e\right. \end{bmatrix}$$

Methods suggested online only provide numerical solutions or partial sums:

- python - Sympy Symbolic Matrix Exponential - Stack Overflow
- python - Exponentiate symbolic matrix expression using SymPy - Stack Overflow
- Calculate state transition matrix in python - Stack Overflow

Instead this will need to be implemented from first principles.

3.1 Theory

3.1.1 Matrix Exponentiation

A Matrix Exponential is defined by using the ordinary exponential power series [**?**, Ch. 2],[**?**, Ch. 8.4] (should we prove the power series generally?):

$$e^{\mathbf{X}} = \sum_{k=0}^{\infty} \left[\frac{1}{k!} \cdot \mathbf{X^k} \right] \tag{1}$$

This definition can be expanded upon however by using properties of logarithms:

$$b = e^{\log_e(b)}, \quad \forall b \in \mathbb{C} \tag{2}$$

$$\implies b^{\mathbf{X}} = \left(e^{\log_e(b)} \right)^{\mathbf{X}} \tag{3}$$

$$\implies b^{\mathbf{X}} = e^{\log_e b\mathbf{X}} \tag{4}$$

The identity in (2) is justified by the definition of the complex log. However some discussion is required for (3) because it is not clear that the exponential will generally distribute throught he parenthesis like so $(a \cdot b)^k = a^k \cdot b^k$, for example consider $\left([-1]^2 \cdot 3 \right)^{\frac{1}{2}} \neq [-1]^{\frac{2}{2}} \cdot 3^{\frac{1}{2}}$.

A sufficient condition for this identity is $k \in \mathbb{Z}^*$, consider this example which will be important later:

$$\left(\log_e(b)\mathbf{X} \right)^k, \quad \forall k \in \mathbb{Z}^* \tag{5}$$

Because multiplication is commutative $\forall z \in \mathbb{C}$, this could be re-expressed in the form:

$$\begin{aligned} \left(\log_e(b)\mathbf{X} \right)^k &= \underbrace{\log_e(b) \cdot \log_e(b) \cdot \log_e(b) \ldots}_{k \text{ times}} \times \underbrace{\mathbf{XXX} \ldots}_{k \text{ times}} \\ &= \log_e^k(b)\mathbf{X}^k \end{aligned} \tag{6}$$

Now consider the the following by applying (6):

$$
\begin{aligned}
e^X &= \sum_{k=0}^{\infty}\left[\frac{1}{k!}\mathbf{X}^k\right] \\
\implies e^{bX} &= \sum_{k=0}^{\infty}\left[\frac{1}{k!}\left(b\mathbf{X}\right)^k\right] \quad \forall b \in \mathbb{C} \\
&= \sum_{k=0}^{\infty}\left[\frac{1}{k!}b^k\mathbf{X}^k\right] \\
&= \left(e^b\right)^{\mathbf{X}} \\
&\implies e^{b\mathbf{X}} = e^{\mathbf{X}b} = \left(e^b\right)^{\mathbf{X}} = \left(e^{\mathbf{X}}\right)^b \qquad \square
\end{aligned} \tag{7}
$$

So the matrix exponential for an arbitrary base could be given by:

$$
\begin{aligned}
b = e^{\log_e(b)}, \quad & \forall b \in \mathbb{C} \\
&\implies b^{\mathbf{X}} = \left(e^{\log_e(b)}\right)^{\mathbf{X}} \\
&\text{as per (7)} \\
&\qquad b^{\mathbf{X}} = e^{\log_e(b)\mathbf{X}} \\
&\qquad b^{\mathbf{X}} = \sum_{k=0}^{\infty}\left[\frac{\left(\log_e(b)\mathbf{X}\right)^k}{k!}\right] \\
&\qquad\quad = \sum_{k=0}^{\infty}\left[\frac{\log_e^k(b)}{k!}\mathbf{X}^k\right]
\end{aligned} \tag{8}
$$

This is also consistent with the *McLaurin Series* expansion of $b^{\mathbf{X}}$ ($\forall b \in \mathbb{C}$):

$$
\begin{aligned}
f(x) &= \sum_{k=0}^{\infty}\left[\frac{f^{(n)}(0)}{k!}x^k\right] \\
\implies b^x &= \sum_{k=0}^{\infty}\left[\frac{\frac{\mathrm{d}^n}{\mathrm{d}x^n}\left(b^x\right)|_{x=0}}{k!}x^k\right] \\
\implies b^{\mathbf{X}} &= \sum_{k=0}^{\infty}\left[\frac{\frac{\mathrm{d}^n}{\mathrm{d}\mathbf{X}^n}\left(b^{\mathbf{X}}\right)|_{\mathbf{X}=\mathbf{O}}}{k!}\mathbf{X}^k\right]
\end{aligned}
$$

By ordinary calculus identities we have$f(x) = b^x \implies f^{(n)}(x) = b^x \log_e^n(b)$ which distribute through a matrix and hence:

$$b^x = \sum_{k=0}^{\infty} \left[\frac{b^0 \log_e^k(b)}{k!} x^k \right]$$
$$\implies b^{\mathbf{X}} = \sum_{k=0}^{\infty} \left[\frac{b^0 \log_e^k(b)}{k!} \mathbf{X}^k \right]$$

By the previous identity:

$$\implies b^{\mathbf{X}} = \sum_{k=0}^{\infty} \left[\frac{(\log_e(b)\mathbf{X})^k}{k!} \right]$$
$$= e^{\log_e(b)\mathbf{X}}$$

3.1.2 Matrix-Matrix Exponentiation

Matrix-Matrix exponentiation has applications in quantum mechanics [**?**, p. 84].

As for Matrices with the requirements:

1. Square
2. Normal:
    - Commutes with it's congugate transpose
3. Non Singular
4. Non Zero Determinant

$$||A - I|| < 1 \implies e^{\log_e(\mathbf{A})} = \mathbf{A} \text{ (By Lie Groups Springer Textbook)}$$
$$\implies \mathbf{A}^{\mathbf{B}} = \left(e^{\log_e(\mathbf{A})}\right)^{\mathbf{B}}$$
Similar justification as (7)
$$\implies \mathbf{A}^{\mathbf{B}} = e^{\log_e(\mathbf{A})\mathbf{B}}$$

However the following identities are by **Definition** anyway: [**?**]

$$\mathbf{A}^{\mathbf{B}} = e^{\log_e(\mathbf{A})\mathbf{B}} \tag{9}$$

$$^{\mathbf{B}}\mathbf{A} = e^{\mathbf{B}\log_e(\mathbf{A})} \tag{10}$$

3.2 An alternative Implementation in Sympy

```
def matexp(mat, base = E):
      """
      Return the Matrix Exponential of a square matrix
      """
      import copy
      import sympy
  # Should realy test for sympy vs numpy array
  # Test for Square Matrix
      if mat.shape[0] != mat.shape[1]:
          print("ERROR: Only defined for Square matrices")
          return
      m = zeros(mat.shape[0])
      for i in range(m.shape[0]):
          for j in range(m.shape[1]):
              m[i,j] = Sum((mat[i,j]*ln(base))**k/factorial(k), (k, 0, oo)).doit()
      return m

matexp(A, pi)
```

$$\begin{bmatrix} \pi^{11} & \pi^{12} & \pi^{13} \\ \pi^{21} & \pi^{22} & \pi^{23} \\ \pi^{31} & \pi^{32} & \pi^{33} \end{bmatrix}$$

But it would be nice to expand this to matrix bases for there uses in quantum mechanics.

The built in method for a**mat is not implemented.

there is exp(mat) but this returns garbage (see github issue), (see other solution on stack exchange that is numeric and example)

show our method with proofs of

cauchy power taylor then exp

then show our code

```
A = Matrix([ [11,12,13], [21,22,23], [31,32,33] ])
```

```
B = Matrix([
    [1,2,3],
    [4,5,6],
    [7,8,9]
])


A**B
```

4 Recursive Relations

A recursive relation is of the form:

$$\sum_{n=0}^{\infty} [c_i \cdot a_n] = 0, \quad \exists c \in \mathbb{R}, \ \forall i < k \in \mathbb{Z}^+$$

In order to find a solution for a_n, the following one-to-one correspondence can be used to relate the vector space of the sequence to the power series ring:(cite stackExchange[1]):

$$g : (a_n)_{n \in \mathbb{Z}^+} \to \mathbb{C}\left[[x]\right] : g\left(a_n\right) = \sum_{n=0}^{\infty} \left[\frac{x^n}{n!} a_n\right]$$

This technique is referred to as generating functions. [**?**]

4.1 TODO Generating Functions

4.1.1 TODO Example

4.2 TODO Exponential Generating Function

4.2.1 TODO Example

4.2.2 TODO Homogoneous Proof

1. **TODO** Derivative of Exponential Generating Function CITE OPEN MIT COURSEWARE SLIDES
2. **TODO** Unique Roots of Characteristic Equation
   (a) Example
   (b) Proof

3. **TODO** Repeated Roots of Characteristic Equation
   (a) Example
   (b) Proof
4. **TODO** General Proof Consider a Recursive relation with constant coefficients:

$$\sum_{n=0}^{\infty} [c_i \cdot a_n] = 0, \quad \exists c \in \mathbb{R}, \ \forall i < k \in \mathbb{Z}^+$$

This can be expressed in terms of the exponential generating function:

$$\sum_{n=0}^{\infty} [c_i \cdot a_n] = 0 \implies \sum_{n=0}^{\infty} \left[\sum_{n=0}^{\infty} [c_i \cdot a_n] \right] = 0$$

4.3 Pandoc Conversion

Given the Linear Recurrence Relation:

$$\begin{aligned} a_0 &= 1 \\ a_0 &= 1 \\ a_{n+2} = a_{n+1+2a_n}, &\quad n \geq 0 \end{aligned}$$

To solve this we can use what's known as a Generating Function, see the disucssion below

We will make consider the function $f(x)$ as shown below in:

$$f(x) = \sum_{n=0}^{\infty} [a_n x^n] \tag{11}$$

It can be shown (see (**??**)) that:

$$\sum_{n=0}^{\infty} [a_{n+1} x^n] = \frac{f(x) - a_0}{x} \tag{12}$$

$$\sum_{n=0}^{\infty} [a_{n+2} x^n] = \frac{f(x) - a_0 - a_1 x}{x^2} \tag{13}$$

So to use the generating Function consider:

$$\begin{aligned} 2a_n + a_{n+1} &= a_{n+2} \\ 2a_n x^n + a_{n+1} x^n &= a_{n+2} x^n \\ \sum_{n=0}^{\infty} [2a_n x^n] + \sum_{n=0}^{\infty} [a_{n+1} x^n] &= \sum_{n=0}^{\infty} [a_{n+2} x^n] \end{aligned} \tag{14}$$

Observe that in (**??**) tuse te
By applying the previous identity shown in (11), (12) and (13):

$$\begin{aligned} 2f(x) + \frac{f(x) - a_0}{x} &= \frac{f(x) - a_0}{-a_1 x} x^2 \\ \implies f(x) &= \frac{1}{1 - x - x^2} \end{aligned} \tag{15}$$

WARNING

I accidently dropped the 2 here, it doesn't matter but it does show that how this could be dealt with

The function $f(x)$ in (15) can be solved by way of a power series, (see for example 11_{Series}), but first it is necessary to use partial fractions to split it up:

By partial fractions it is known:

$$
\begin{aligned}
f(x) &= \frac{1}{1-x-x^2} \\
&= \frac{-1}{x^2+x-1} \\
&= \frac{-1}{(x-2)(x-1)} \\
&= \frac{A_1}{x-2} + \frac{A_2}{x-1}, \quad A_i \in \mathbb{R}, i \in \mathbb{Z}^+ \\
\implies -1 &= A_1(x-1) + A_2(x-2)
\end{aligned}
$$

Let $x = 2$:

$$
\begin{aligned}
-1 &= A_1(2-1) + 0 \\
&= A_1 = -1
\end{aligned}
$$

Let $x = 1$:

$$
\begin{aligned}
-1 &= 0 + A_2(1-2) \\
\implies A_2 &= 1
\end{aligned}
$$

Hence:

$$
f(x) = \frac{1}{x-1} - \frac{1}{x-2}
$$

Now because it is known that:

$$
\sum_{n=0}^{\infty} [rx^n] = \frac{1}{1-rx^n}
$$

we can conclude that:

$$
\begin{aligned}
\frac{1}{x-1} &= -\frac{1}{1-(1)x} \\
&= -\sum_{n=0}^{\infty} [x^n] \\
\frac{-1}{x-2} &= \frac{1}{2-x} \\
&= \frac{1}{2}\frac{1}{1-\frac{1}{2}x} \\
&= \frac{1}{2}\sum_{n=0}^{\infty}\left[\left(\frac{1}{2}x\right)^n\right]
\end{aligned}
$$

and so:

$$
\begin{aligned}
f(x) &= \frac{1}{2}\sum_{n=0}^{\infty}\left[\left(\frac{1}{2}x\right)^n\right] - \sum_{n=0}^{\infty}\left[x^n\right] \\
f(x) &= \sum_{n=0}^{\infty}\left[\frac{1}{2}\left(\frac{1}{2}x\right)^n - x^n\right] \\
f(x) &= \sum_{n=0}^{\infty}\left[\frac{1}{2\cdot 2^n}x^n - x^n\right] \\
f(x) &= \sum_{n=0}^{\infty}\left[x^n\left(\frac{1}{2\cdot 2^n} - 1\right)\right] \\
\implies a_n &= \frac{1}{2\cdot 2^n} - 1
\end{aligned}
$$

4.3.1 Generating Functions

A Generating Function is a way of encoding an infinite series of numbers (a_n) by treating them as the coeficcients of a power series $(\sum_{n=0}^{\infty}[a_n x^n])$.

The variable remains in an indeterminate form and they were first introduced by Abraham De Moivre in 1730 in order to solve the general linear recurrence problem [1]

[1]: Donald E. Knuth, The Art of Computer Programming, Volume 1 Fundamental Algorithms (Third Edition) Addison-Wesley. ISBN 0-201-89683-4. Section 1.2.9: "Generating Functions".

4.3.2 Using the Power series for the Exponential Function

1. Motivation Consider the *Fibonacci Sequence*:

$$
\begin{aligned}
a_n &= a_{n-1} + a_{n-2} \\
\iff a_{n+2} &= a_{n+1} + a_n
\end{aligned}
$$

Solving this outright is quite difficult, previously we used a power series generating function to solve it, something to the effect of:

$$x^2 f(x) - x f(x) - f(x) = 0$$

This however is still a little tricky, however, just from observation, the following would be fairly easy to deal with:

$$f''(x) - f'(x) - f(x) = 0$$

This would however imply that $f(x) = e^x$ because $\frac{\mathrm{d}(e^x)}{\mathrm{d}x} = e^x$, but that's fine because we have a power series for that already:

$$f(x) = e^r x = \sum_{n=0}^{\infty} \left[r \frac{x^n}{n!} \right]$$

So this would give an easy means by which to solve the linear recurrence relation.

2. Solving the Sequence Now this is all well and good but if we could relate this to $f(x) = e^x$ we'd really be cooking with fire because we could connect linear recurrence relations to non-homogenous linear differential equations.

   Consider using the following generating function:

$$f(x) = \sum_{n=0}^{\infty} \left[a_n \cdot \frac{x^n}{n!} \right] \qquad = e^x$$

   $\mathcal{TODO}$ :: The real trick is showing this derivative property

$$f'(x) = \sum_{n=0}^{\infty} \left[a_{n+1} \cdot \frac{x^n}{n!} \right] \quad = e^x$$

$$f''(x) = \sum_{n=0}^{\infty} \left[a_{n+2} \cdot \frac{x^n}{n!} \right] \quad = e^x$$

   So the recursive relation from above could be expressed:

$$
\begin{aligned}
a_{n+2} &= a_{n+1} + a_n \\
\frac{x^n}{n!} a_{n+2} &= \frac{x^n}{n!} \left(a_{n+1} + a_n \right) \\
\sum_{n=0}^{\infty} \left[\frac{x^n}{n!} a_{n+2} \right] &= \sum_{n=0}^{\infty} \left[\frac{x^n}{n!} a_{n+1} \right] + \sum_{n=0}^{\infty} \left[\frac{x^n}{n!} a_n \right] \\
f''(x) &= f'(x) + f(x)
\end{aligned}
$$

Using the theory of higher order linear differential equations with constant coefficients it can be shown:

$$
f(x) = c_1 \cdot \exp\left[\left(\frac{1 - \sqrt{5}}{2} \right) x \right] + c_2 \cdot \exp\left[\left(\frac{1 + \sqrt{5}}{2} \right) \right]
$$

By equating this to the power series:

$$
f(x) = \sum_{n=0}^{\infty} \left[\left(c_1 \left(\frac{1 - \sqrt{5}}{2} \right)^n + c_2 \cdot \left(\frac{1 + \sqrt{5}}{2} \right)^n \right) \cdot \frac{x^n}{n} \right]
$$

Now given that:

$$
f(x) = \sum_{n=0}^{\infty} \left[a_n \frac{x^n}{n!} \right]
$$

We can conclude that:

$$
a_n = c_1 \cdot \left(\frac{1 - \sqrt{5}}{2} \right)^n + c_2 \cdot \left(\frac{1 + \sqrt{5}}{2} \right)
$$

By applying the initial conditions:

$$
\begin{aligned}
a_0 = c_1 + c_2 &\implies c_1 = -c_2 \\
a_1 = c_1 \left(\frac{1 + \sqrt{5}}{2} \right) - c_1 \frac{1 - \sqrt{5}}{2} &\implies c_1 = \frac{1}{\sqrt{5}}
\end{aligned}
$$

And so finally we have:

$$\begin{aligned} a_n &= \frac{1}{\sqrt{5}}\left[\left(\frac{1+\sqrt{5}}{2}\right)^n - \left(\frac{1-\sqrt{5}}{2}\right)^n\right] \\ &= \frac{\varphi^n - \psi^n}{\sqrt{5}} \\ &= \frac{\varphi^n - \psi^n}{\varphi - \psi} \end{aligned}$$

where:

- $\varphi = \frac{1+\sqrt{5}}{2} \approx 1.61\ldots$
- $\psi = 1 - \varphi = \frac{1-\sqrt{5}}{2} \approx 0.61\ldots$

Open Questions:

- Show that the derivitive of the power series is a_{n+2}
- Redo the initial problem for the Fibonacci Sequence
- Extend this to a non-homogenous equation
- Extend this to all linear recursion problems with first order ODES
- Show that this is an isomorphism Lindear ODEs with constant coefficients to recursive relations with constant coefficients.

4.3.3 References

1. https://math.stackexchange.com/a/1775226
2. https://math.stackexchange.com/a/593553
3. https://www.maa.org/sites/default/files/pdf/upload_library/22/Ford/IvanNiven.pdf

Misc

1. https://brilliant.org/wiki/generating-functions-solving-recurrence-relations/
2. https://www.math.cmu.edu/~af1p/Teaching/Combinatorics/Slides/Generating-Functions.pdf
3. https://www.math.cmu.edu/~af1p/Teaching/Combinatorics/Slides/Generating-Functions.pdf