

METIS

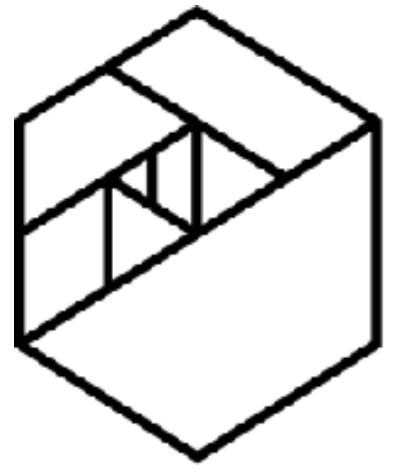
# Introduction to Data Science

Charles S. Givre

@cgivre

[linkedin.com/in/cgivre](https://www.linkedin.com/in/cgivre)

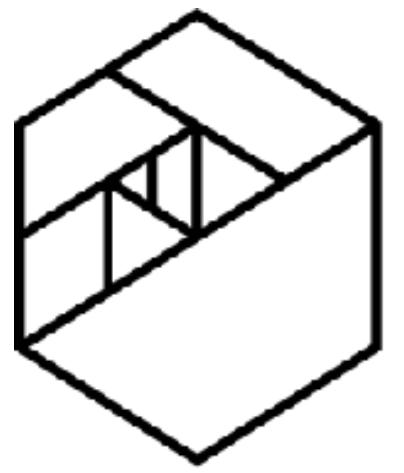
[cgvre@thedataist.com](mailto:cgvre@thedataist.com)



METIS

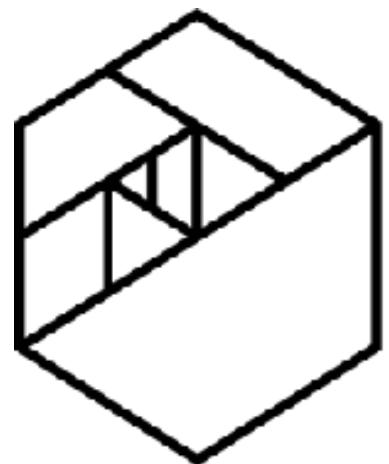
# Expectations

- Please participate and **ask questions**.
- Please follow along and **TRY OUT** the examples yourself during the class
- All the answers are in the slide decks or GitHub repository, but please try to complete the exercises **without looking at the answers**.
- Have fun!



**METIS**

# Introduction



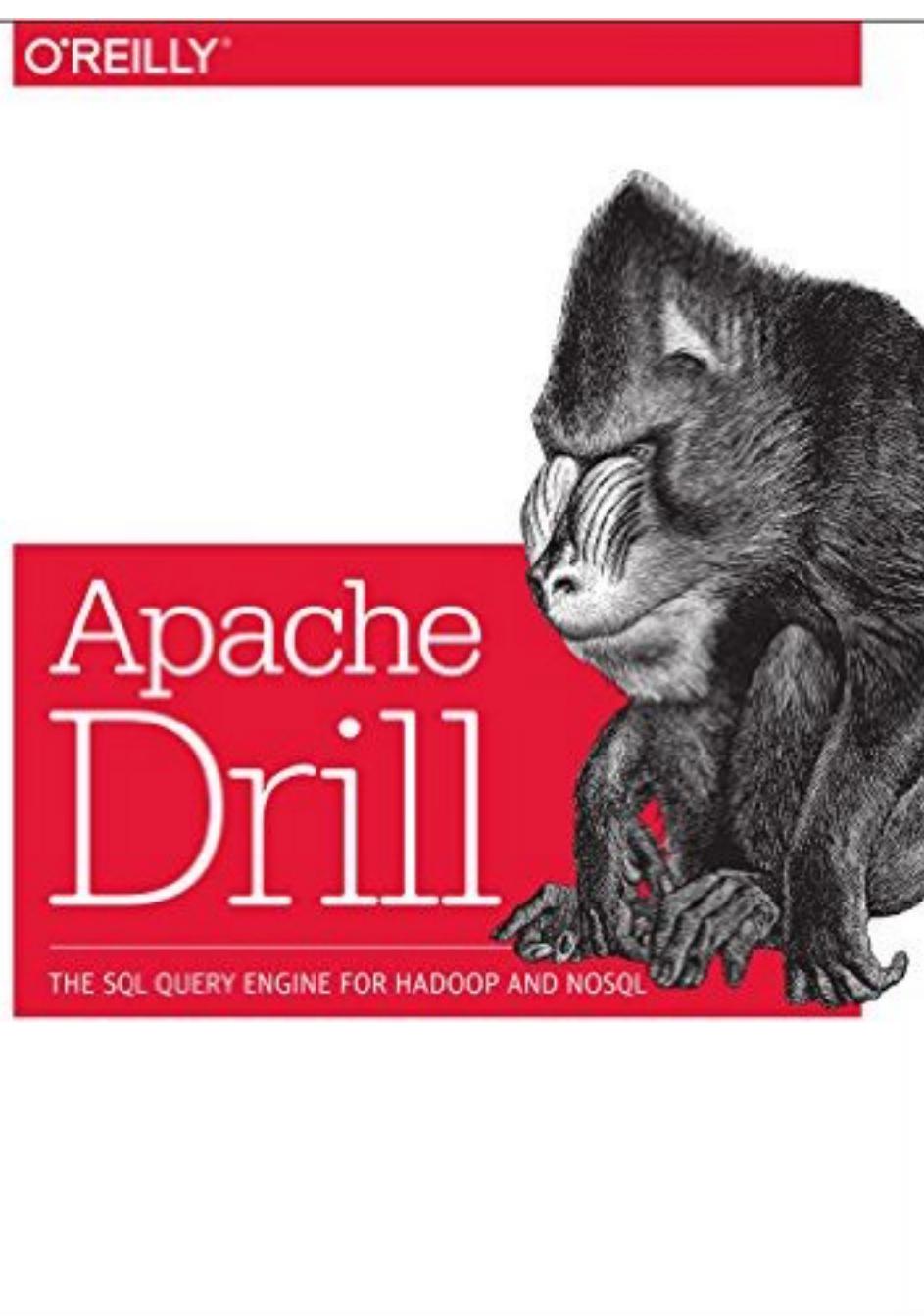
METIS

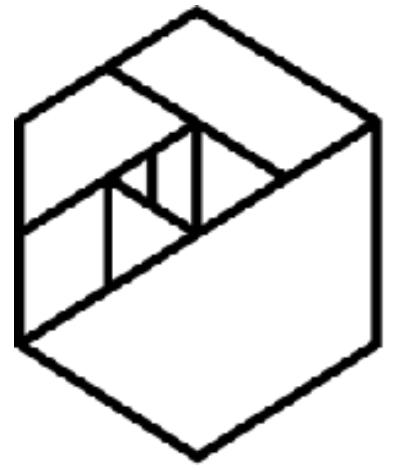
# Charles S. Givre

- 6.5 years Senior Lead Data Scientist @ Booz Allen
- 5 Years @ CIA
- Working on Apache Drill Book
- Masters Degree in Middle Eastern Studies
- Undergraduate in Comp.Sci & Music



Booz | Allen | Hamilton  
100 YEARS

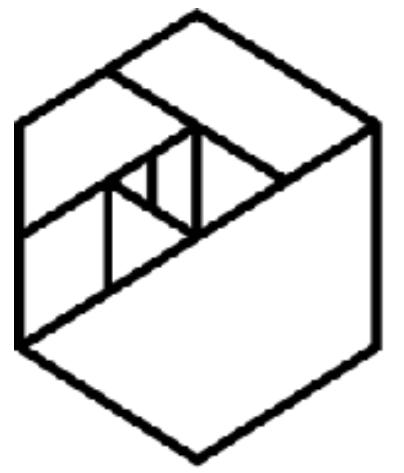




**METIS**

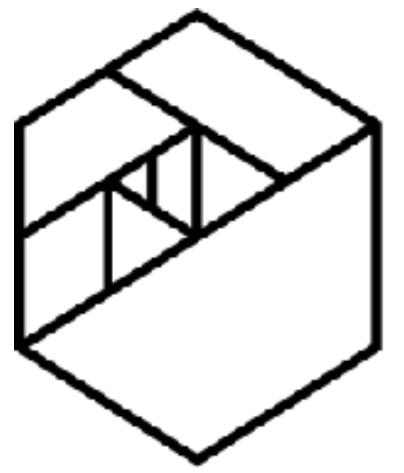
# Who are you?

- Your name (or what you want me to call you)
- Where you work and/or your job role
- What you hope to get out of this class
- Your level of experience with coding



**METIS**

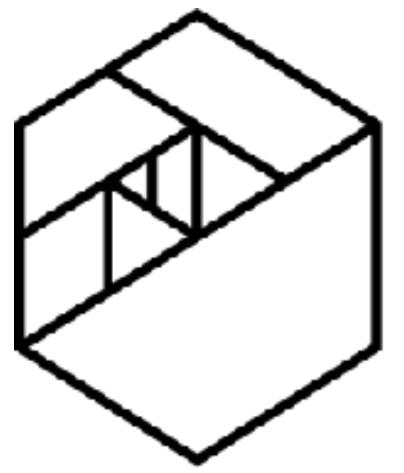
# What is Data Science?



**METIS**

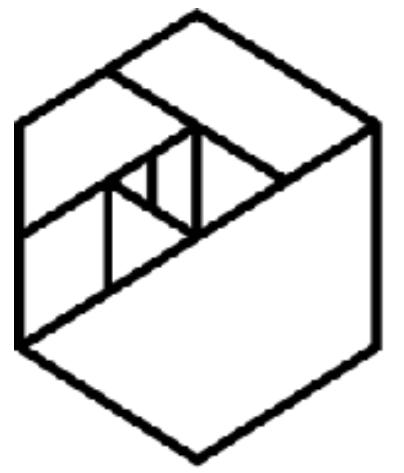
**Data Science is the art of turning data into actions. This is accomplished through the creation of data products, which provide actionable information without exposing decision makers to the underlying data or analytics**

Booz Allen Hamilton, Field Guide to Data Science, Pg. 17



METIS

# Extracting useful information



METIS

# Extracting useful information from data



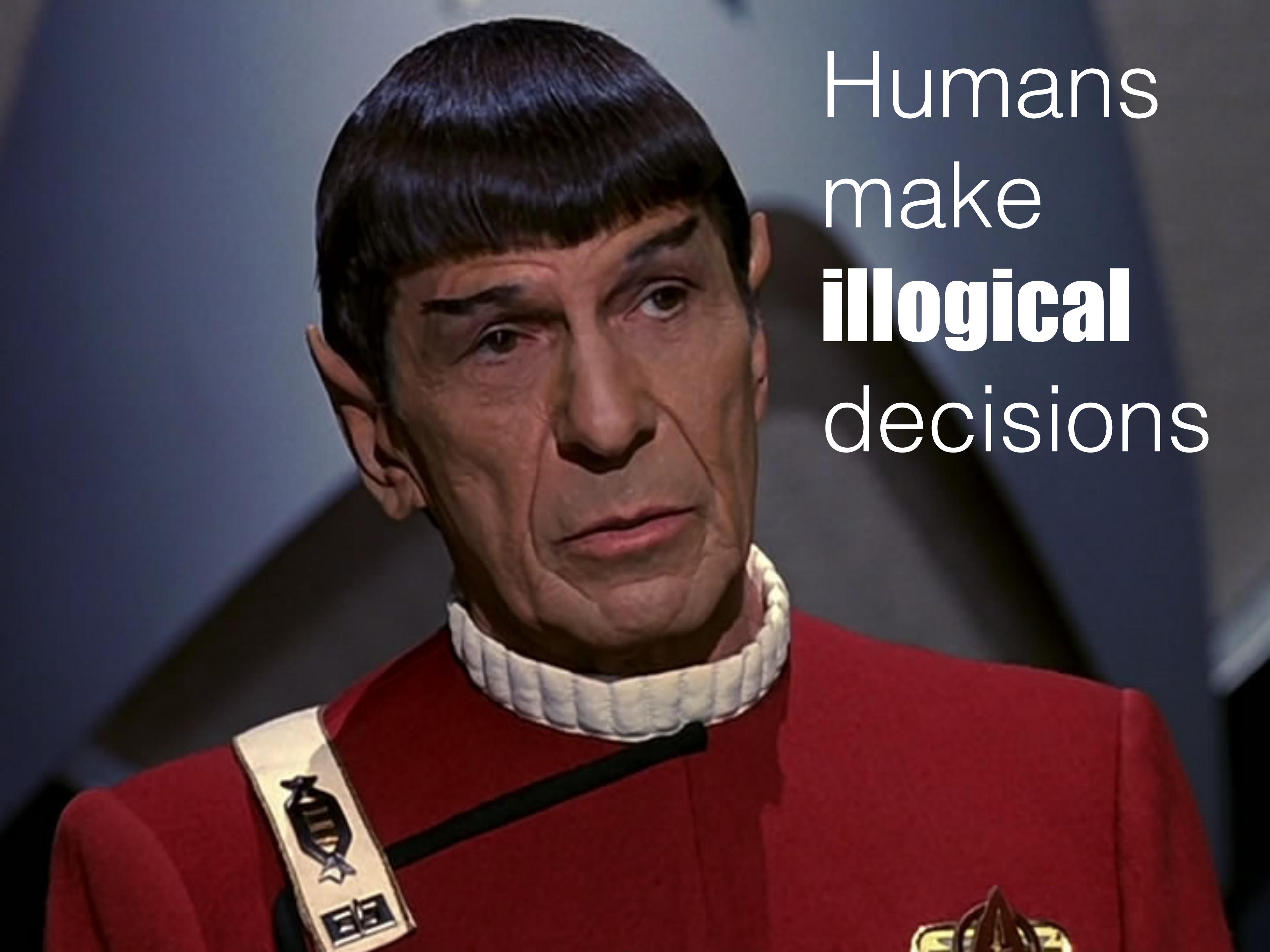
# Answering business questions with data

- Know what you want to know

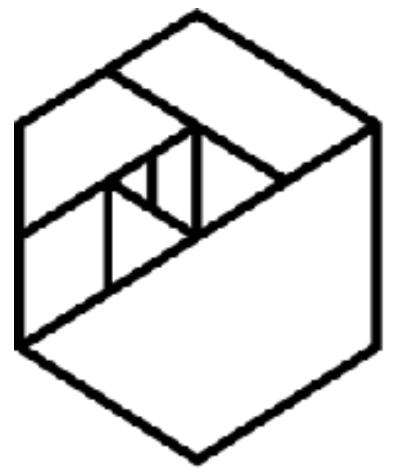


# Answering business questions **with data**

- Know what you want to know
- Have the **technical skills** to get it

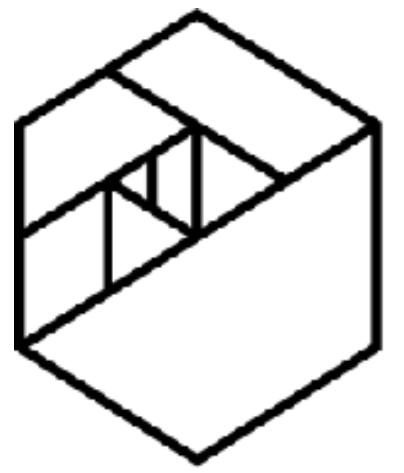


Humans  
make  
**illogical**  
decisions



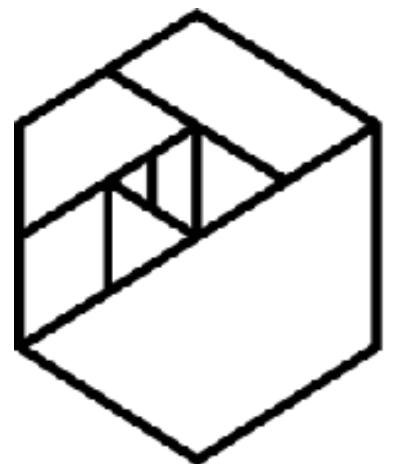
**METIS**

Analyst ← → Developer



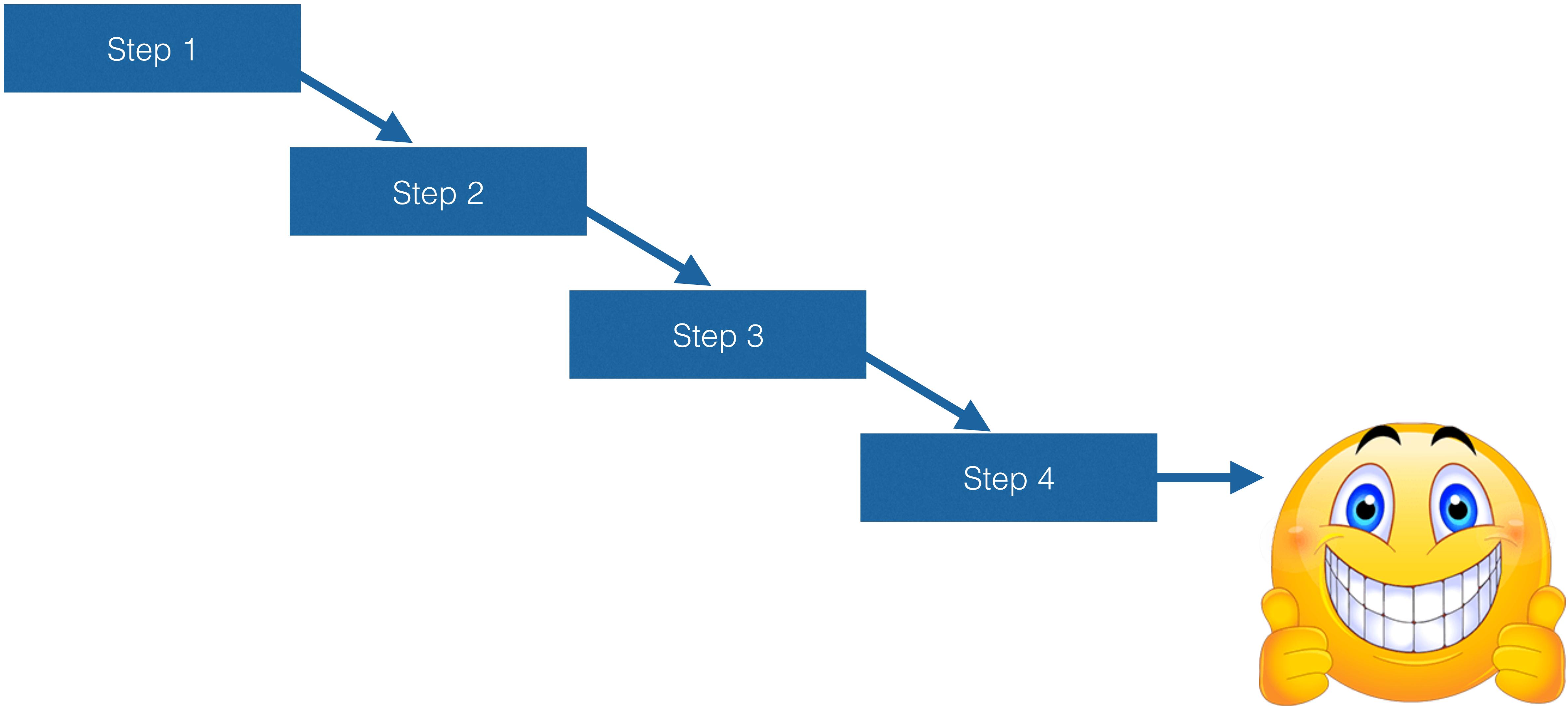
**METIS**

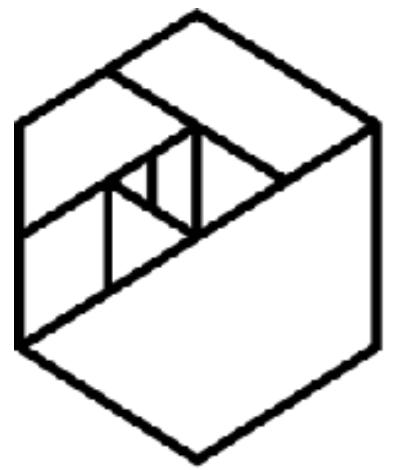
Analyst + Developer



METIS

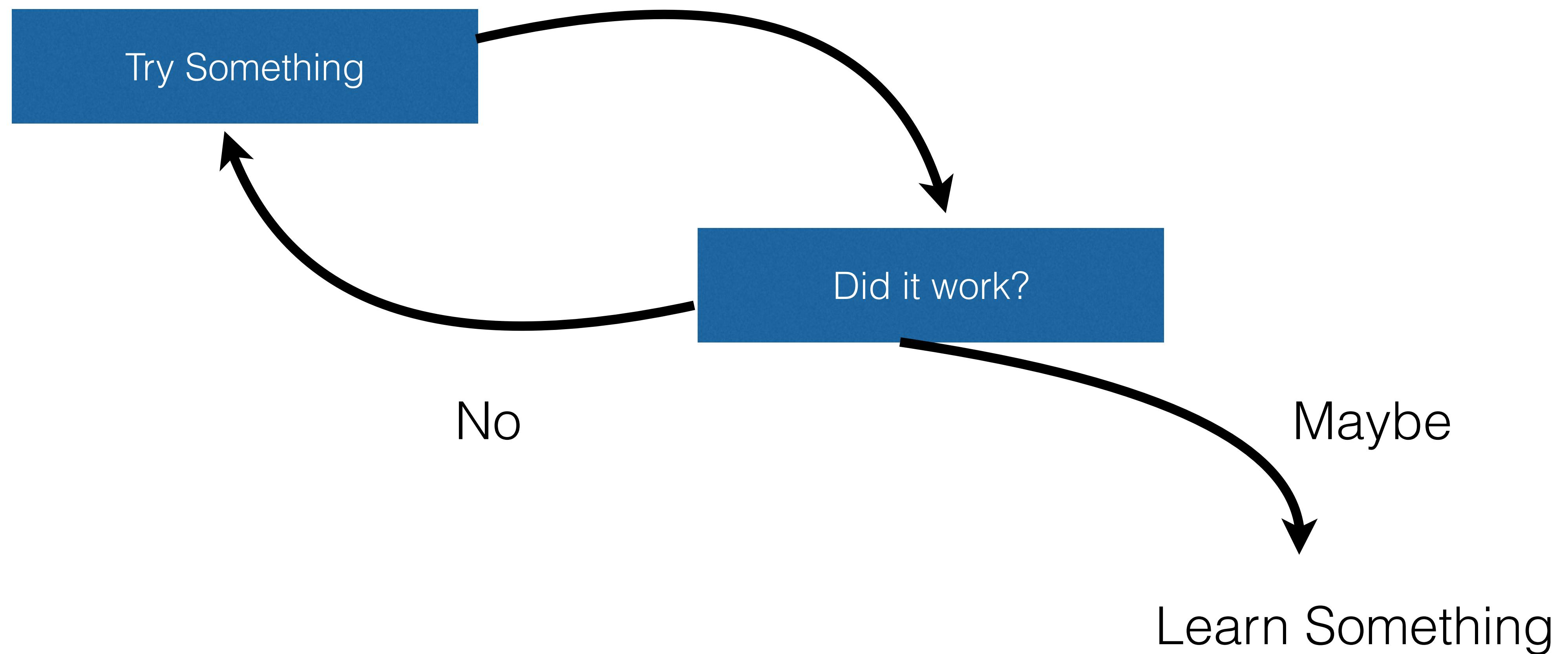
# What Data Science is Not

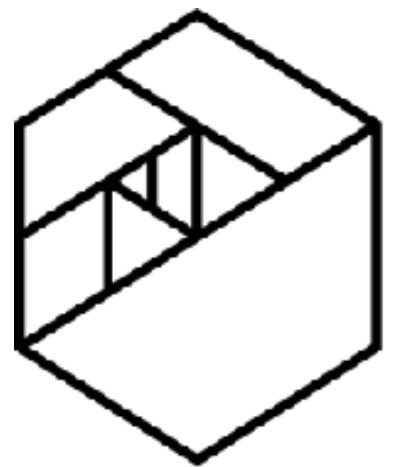




METIS

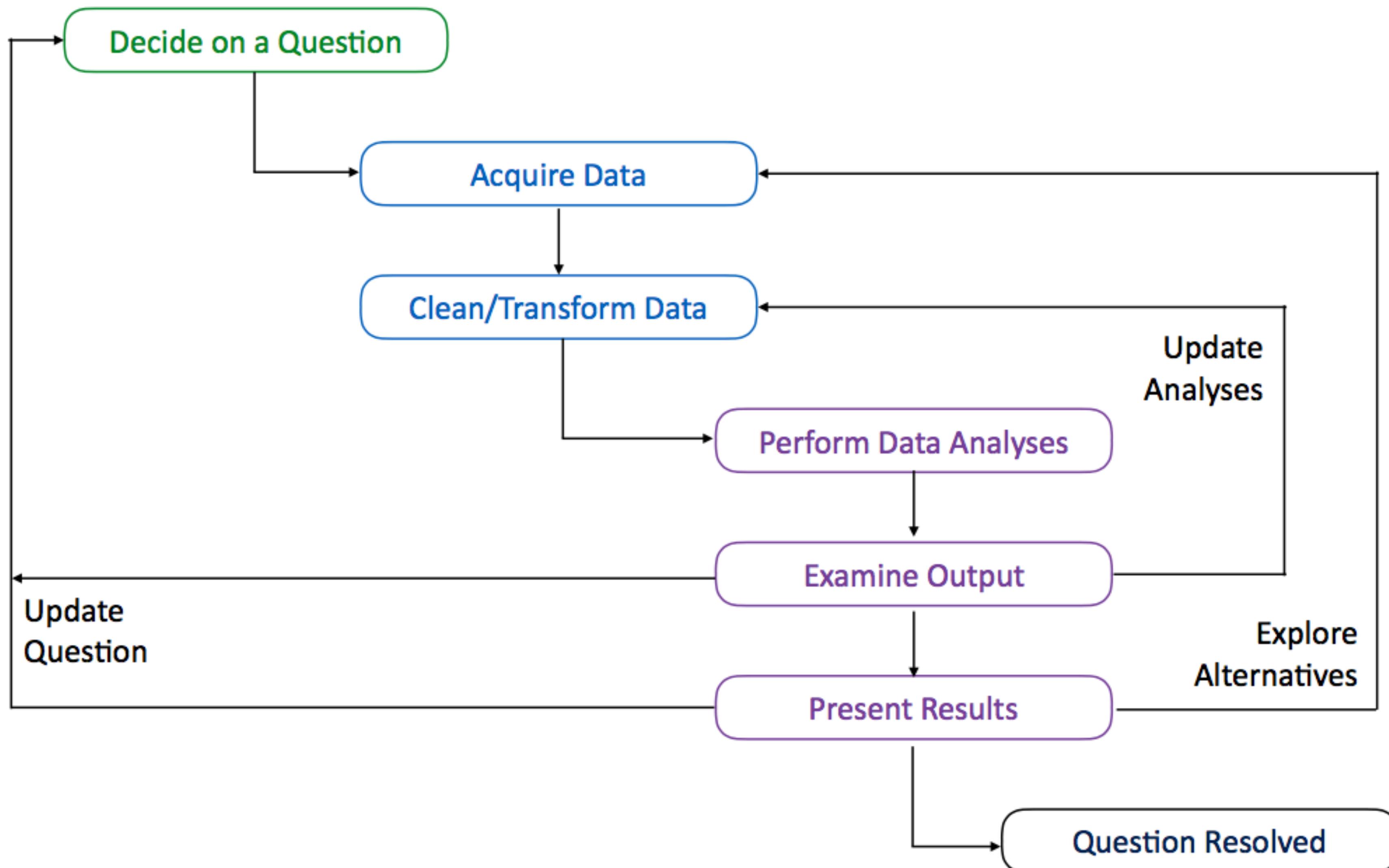
# What Data Science is

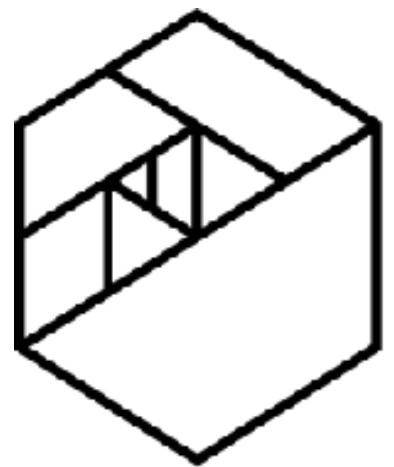




METIS

# Research Process

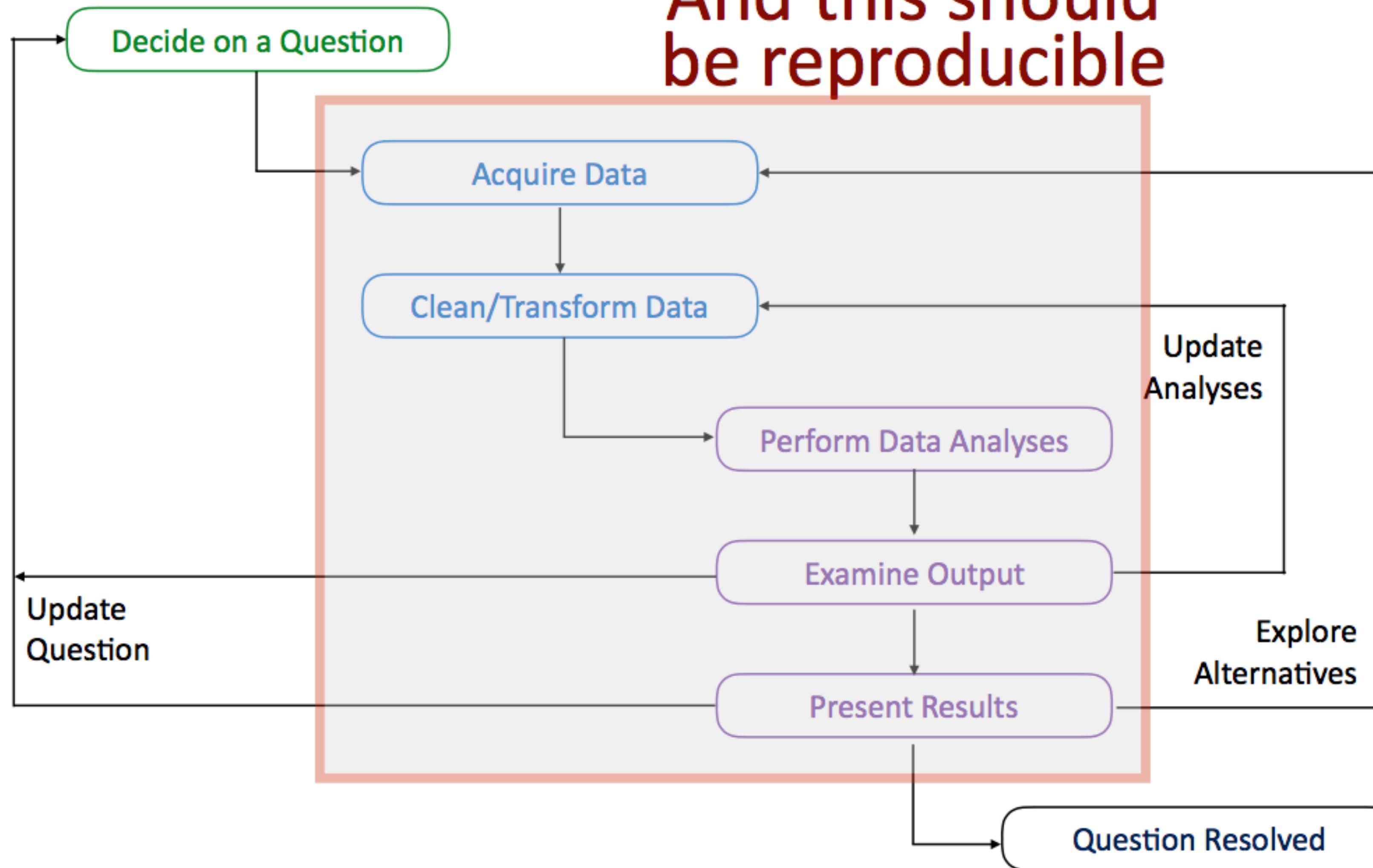


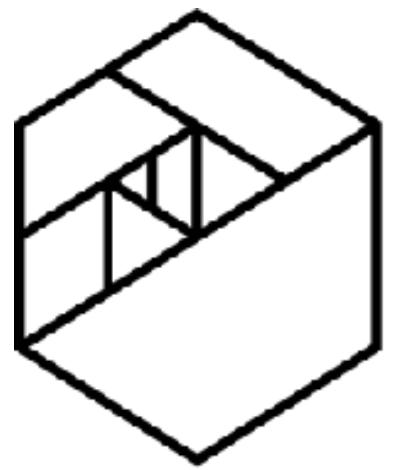


METIS

# Research Process

And this should  
be reproducible

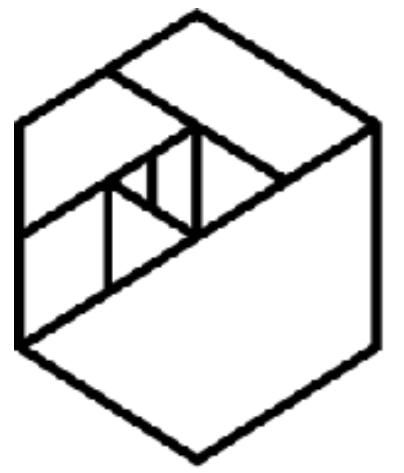




METIS

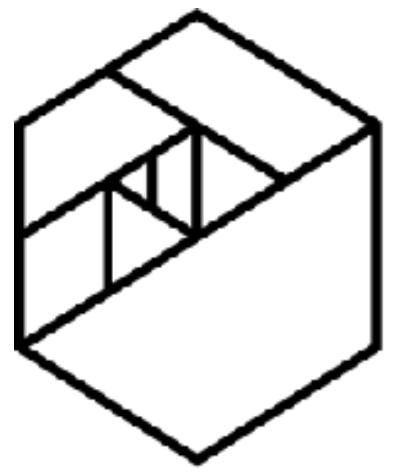
“The term "data scientist" will subside and may well sound dated five years from now. **The skills will become more commonplace and commoditized. When that happens, the real boom will begin, because the technology will become widely adopted and thus more useful. ... Instead, we need self-service tools that empower smart and tenacious business people to perform Big Data analysis themselves.**

–Andrew Brust, “Data scientists don't scale”, <http://www.zdnet.com/article/data-scientists-dont-scale/>



**METIS**

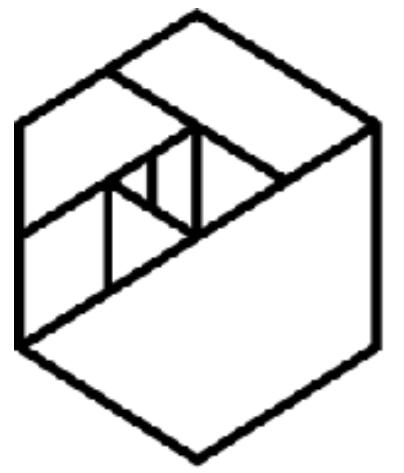
**Time to Insight**



**METIS**

**Time to Insight**

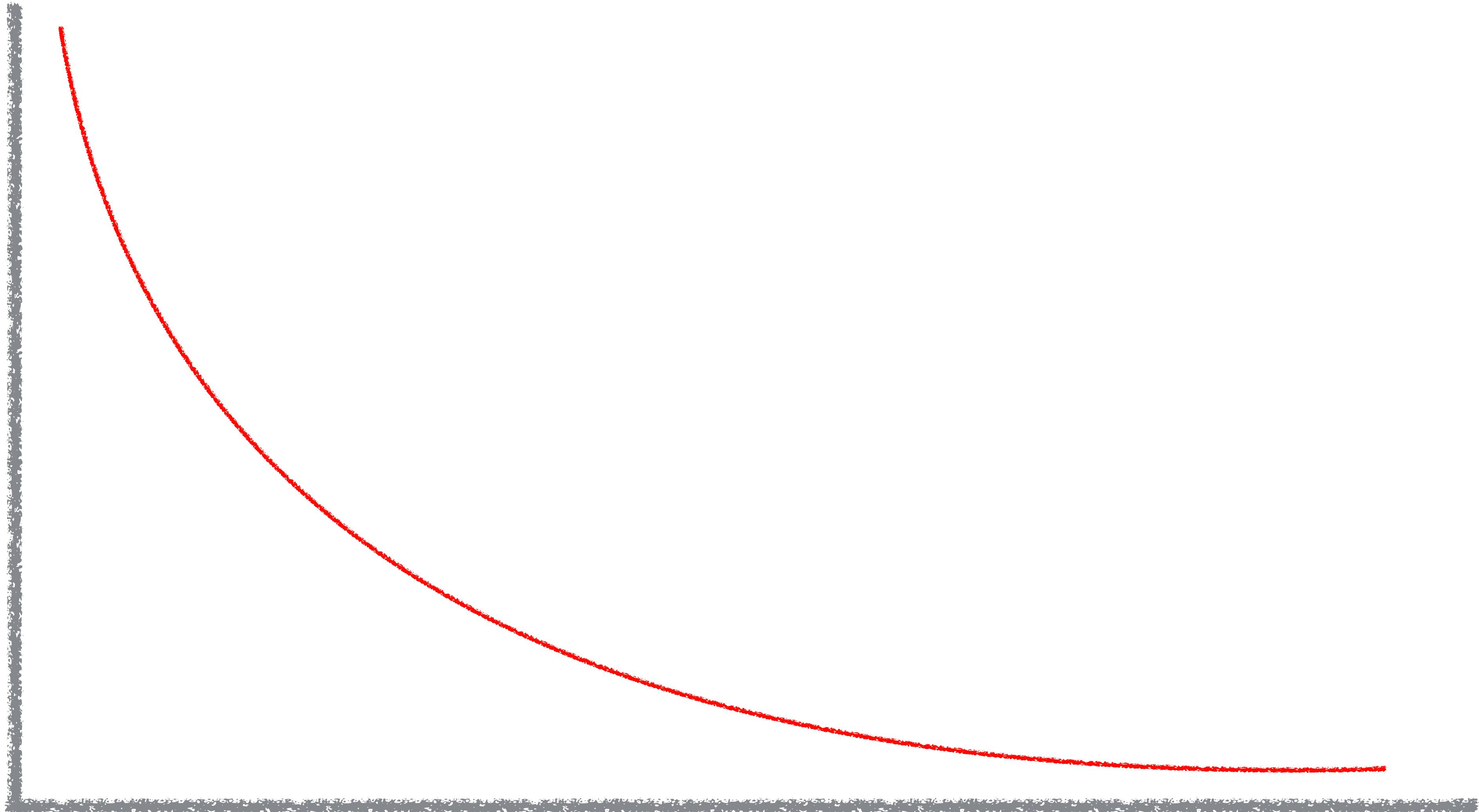
**Time = \$\$**



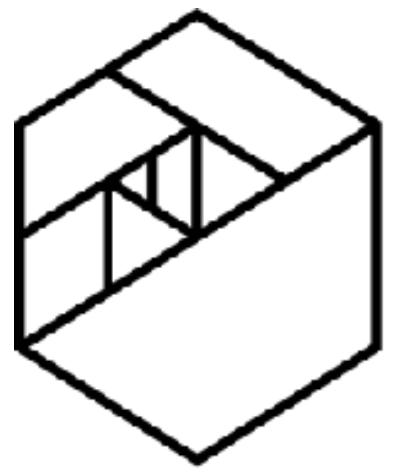
METIS

# Value of Insights over Time

Value



Time

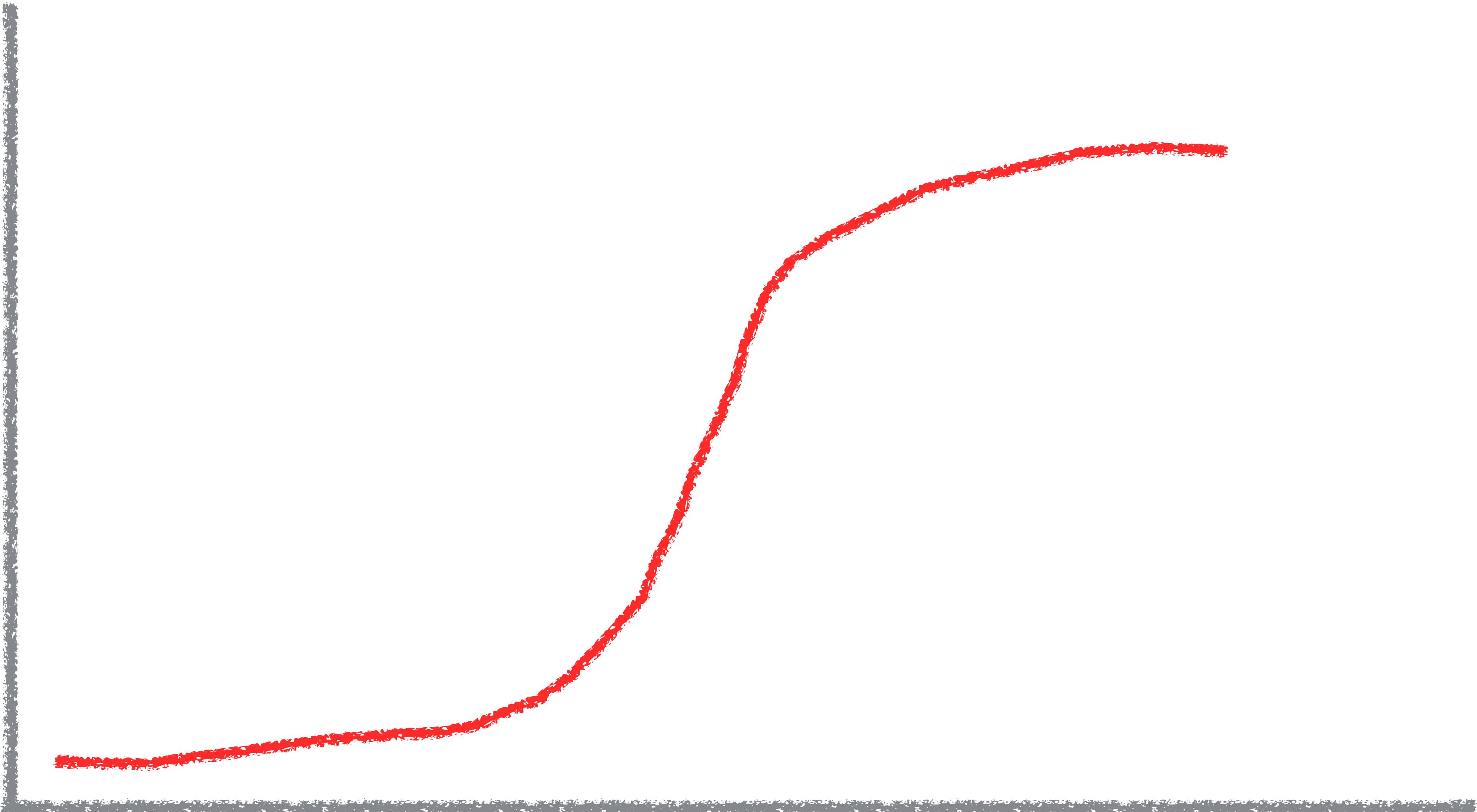


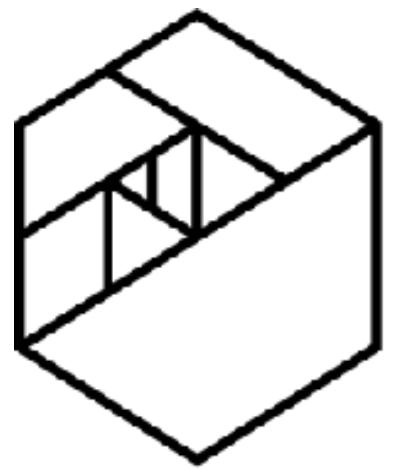
METIS

# Cost of Insights over Time

Cost

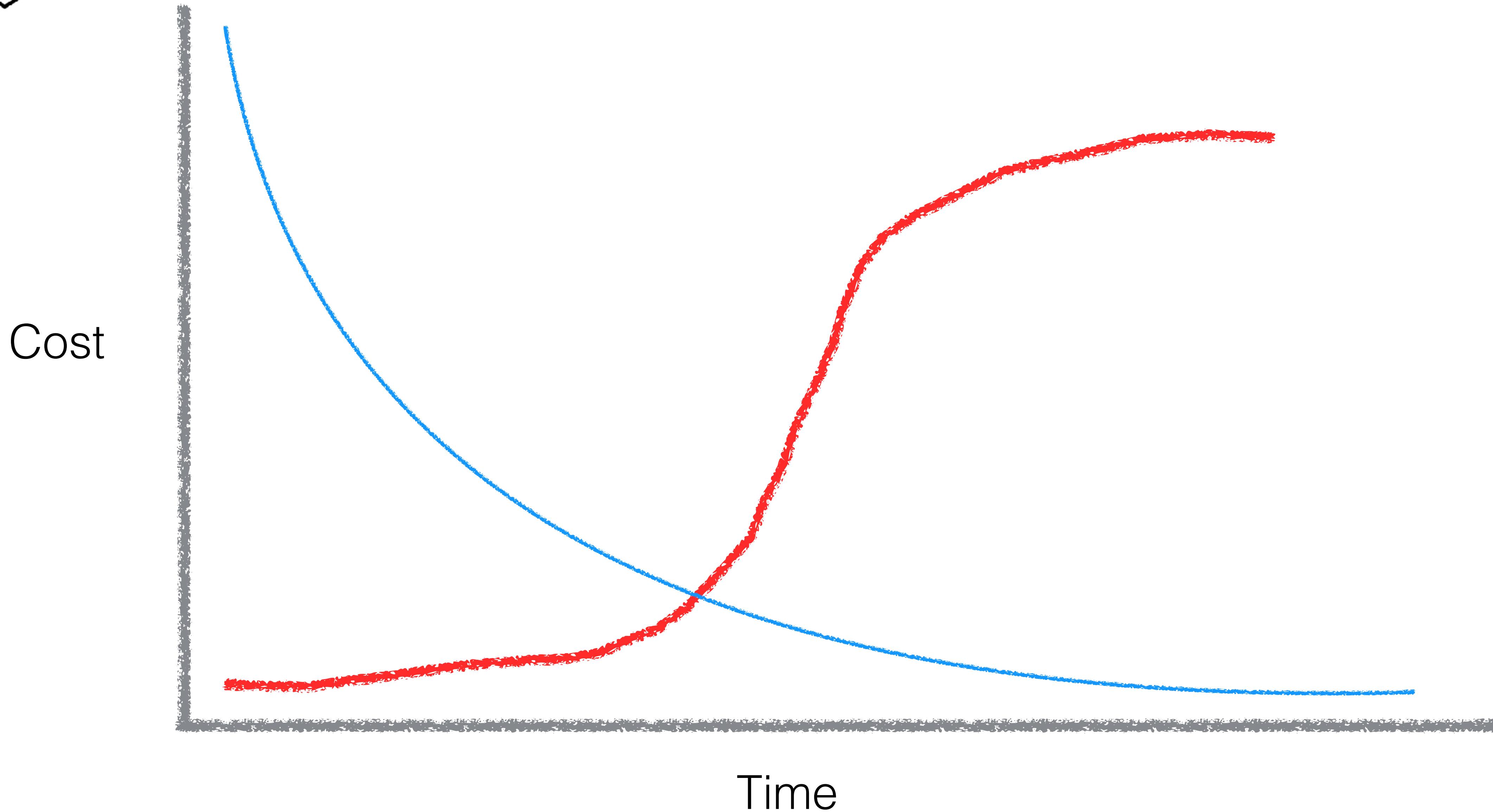
Time

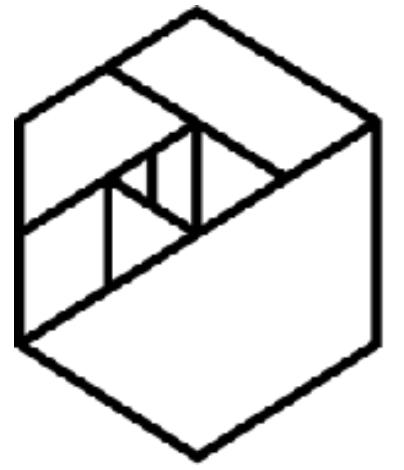




**METIS**

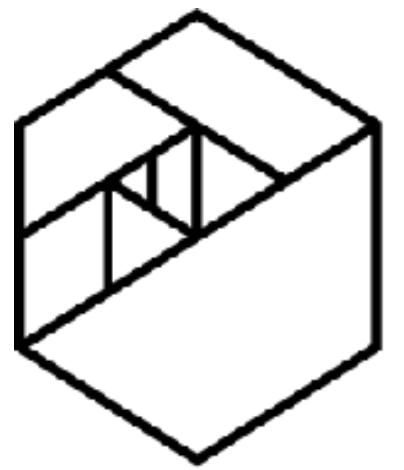
# Cost of Insights over Time



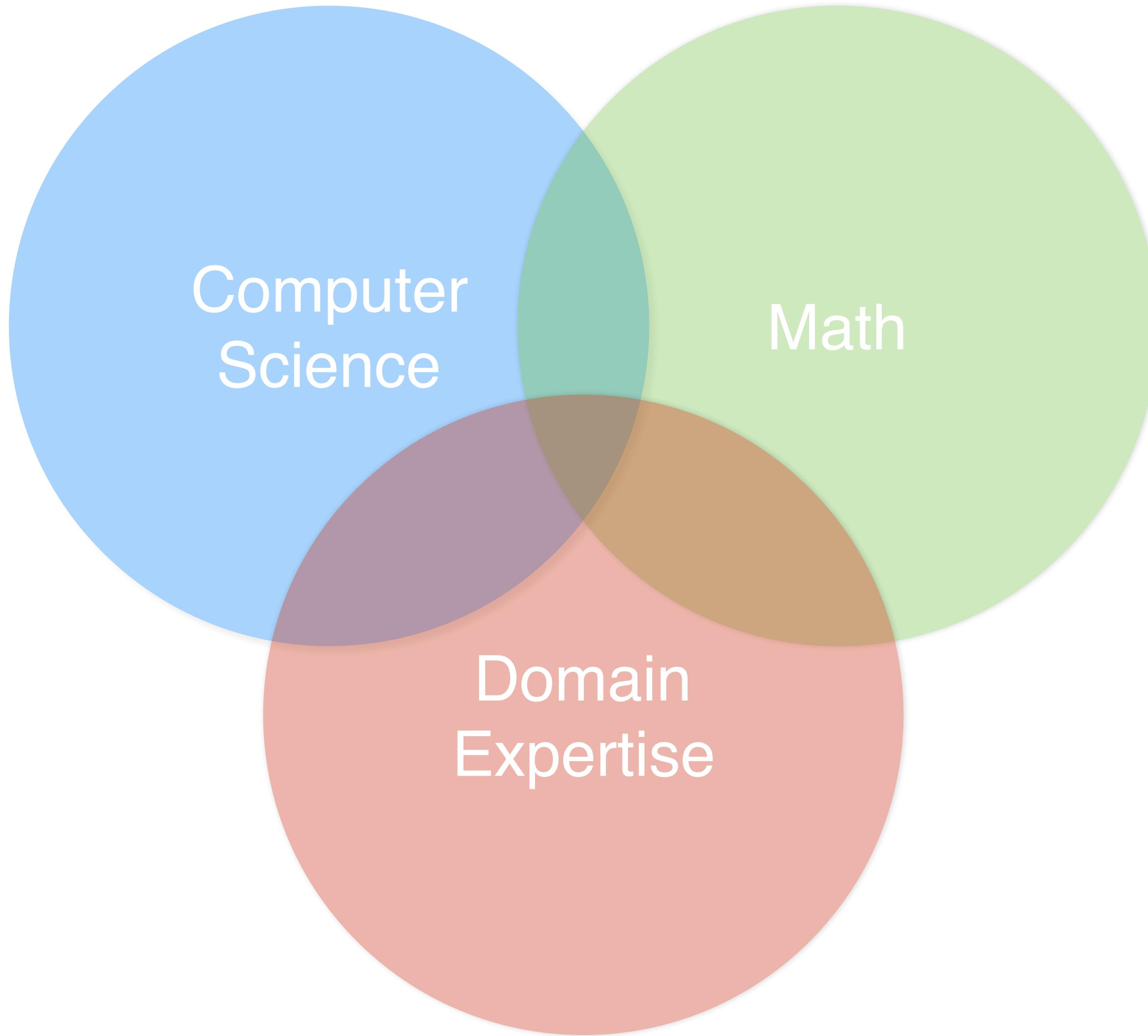


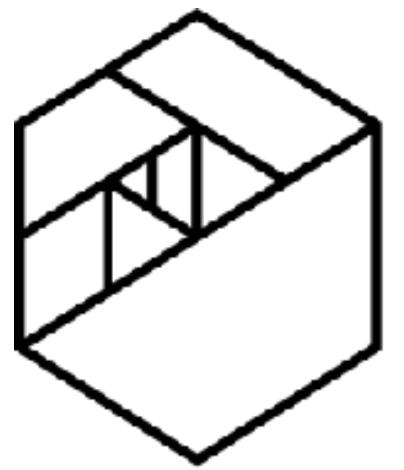
**METIS**

**What skills does a data  
scientist need?**

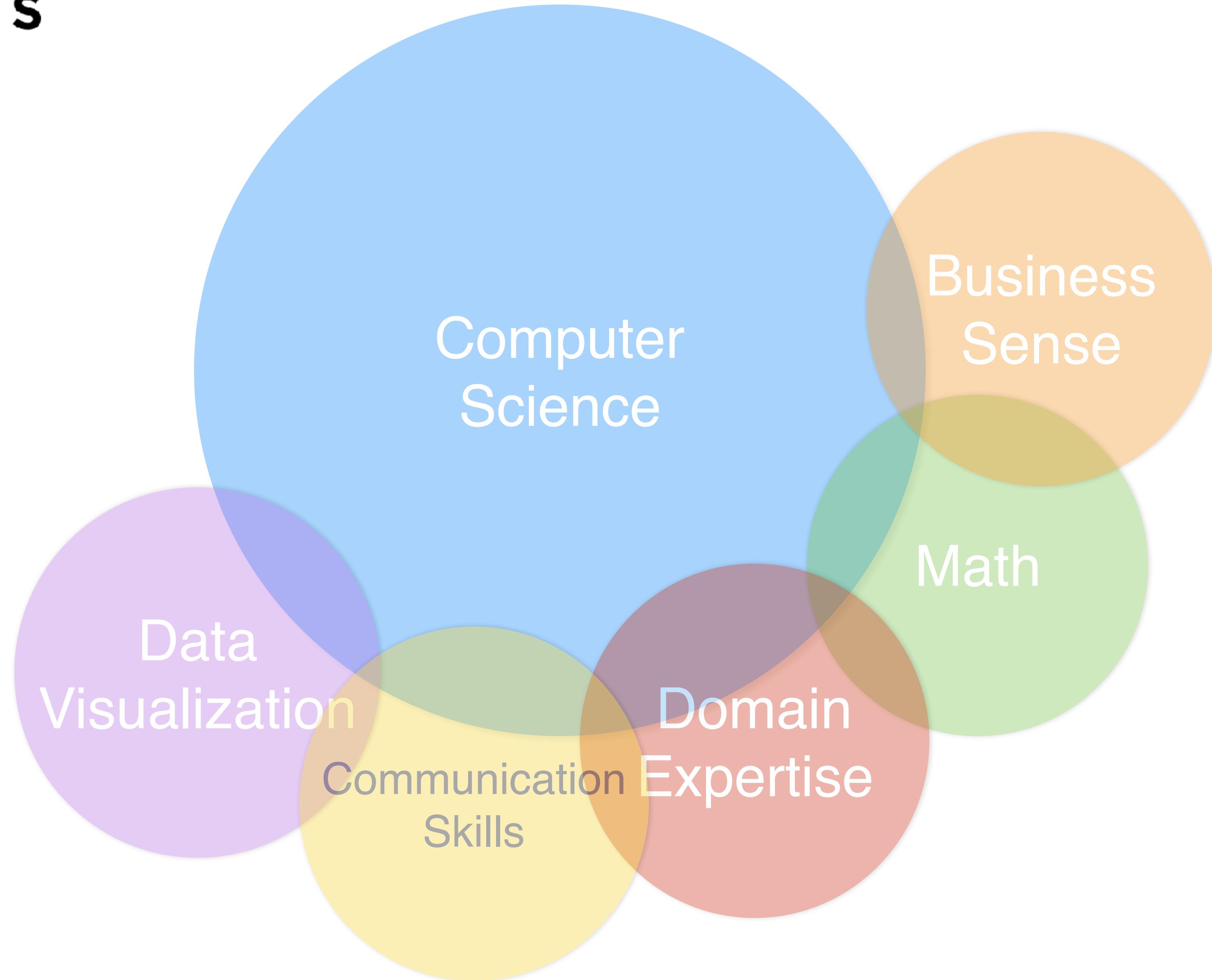


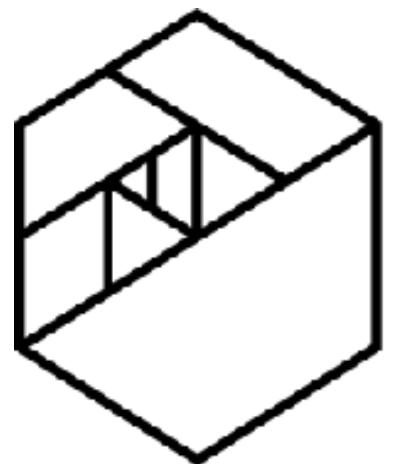
METIS





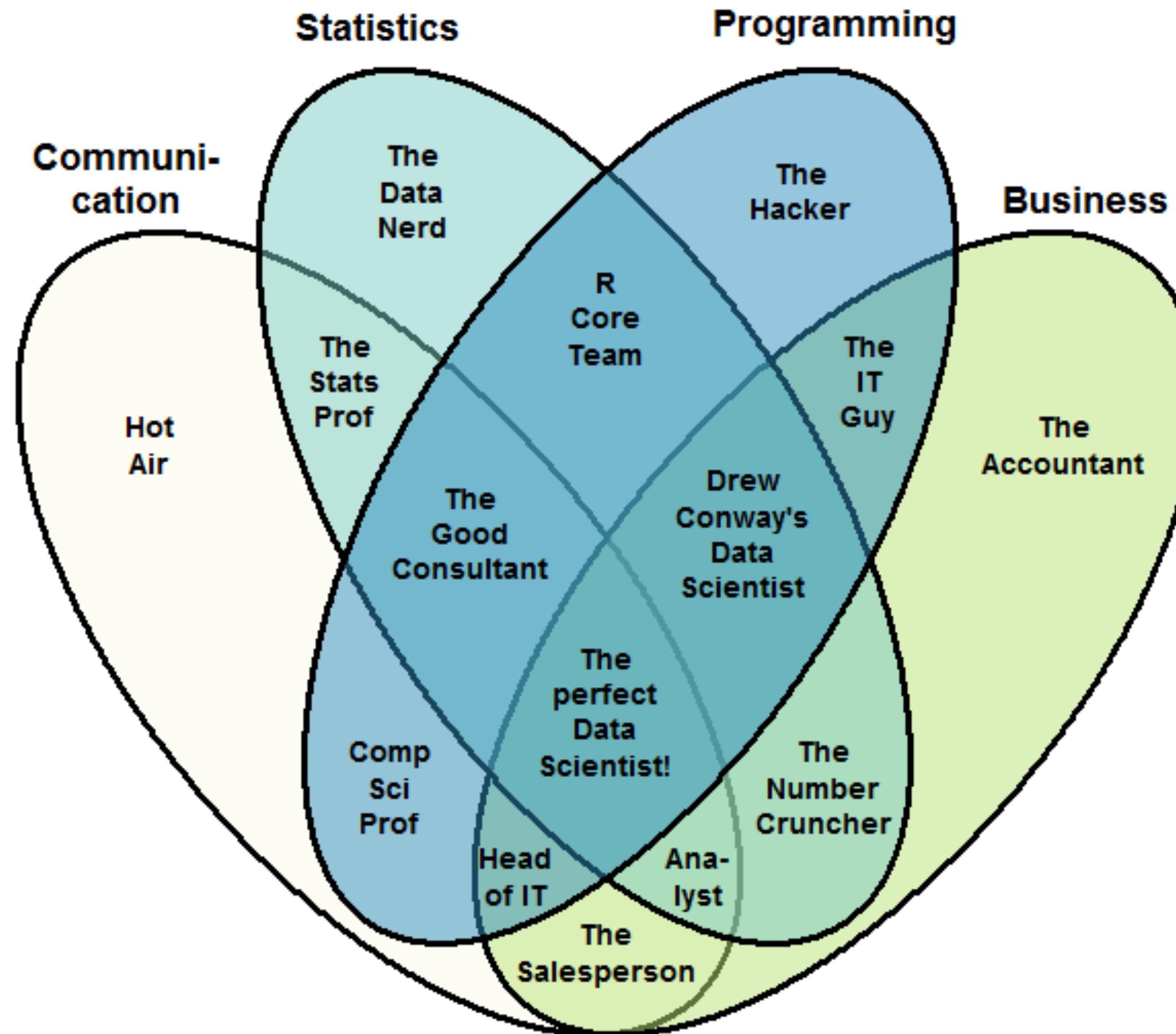
METIS

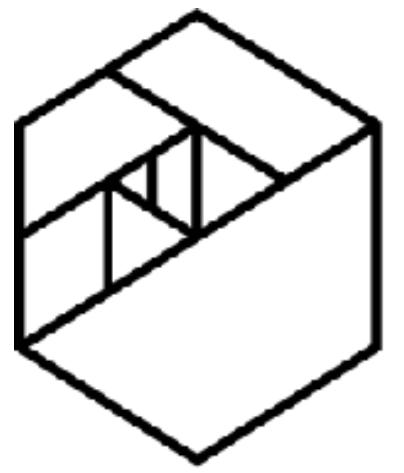




METIS

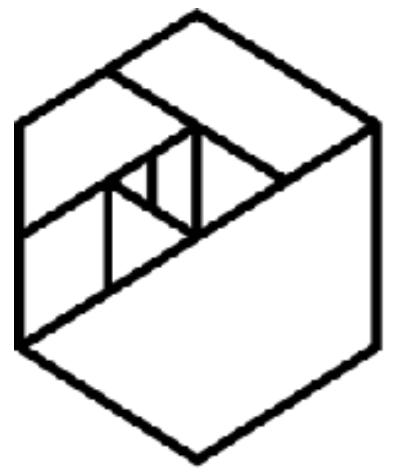
## The Data Scientist Venn Diagram





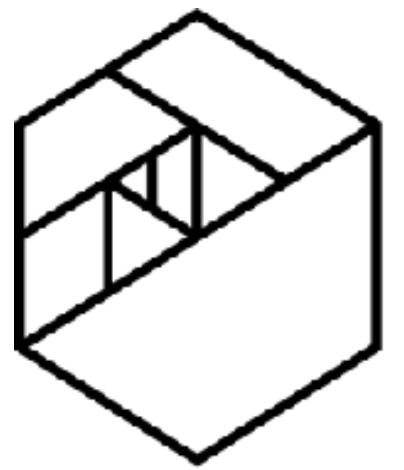
**METIS**

**Data Scientists spend  
50-90% of their time being...**

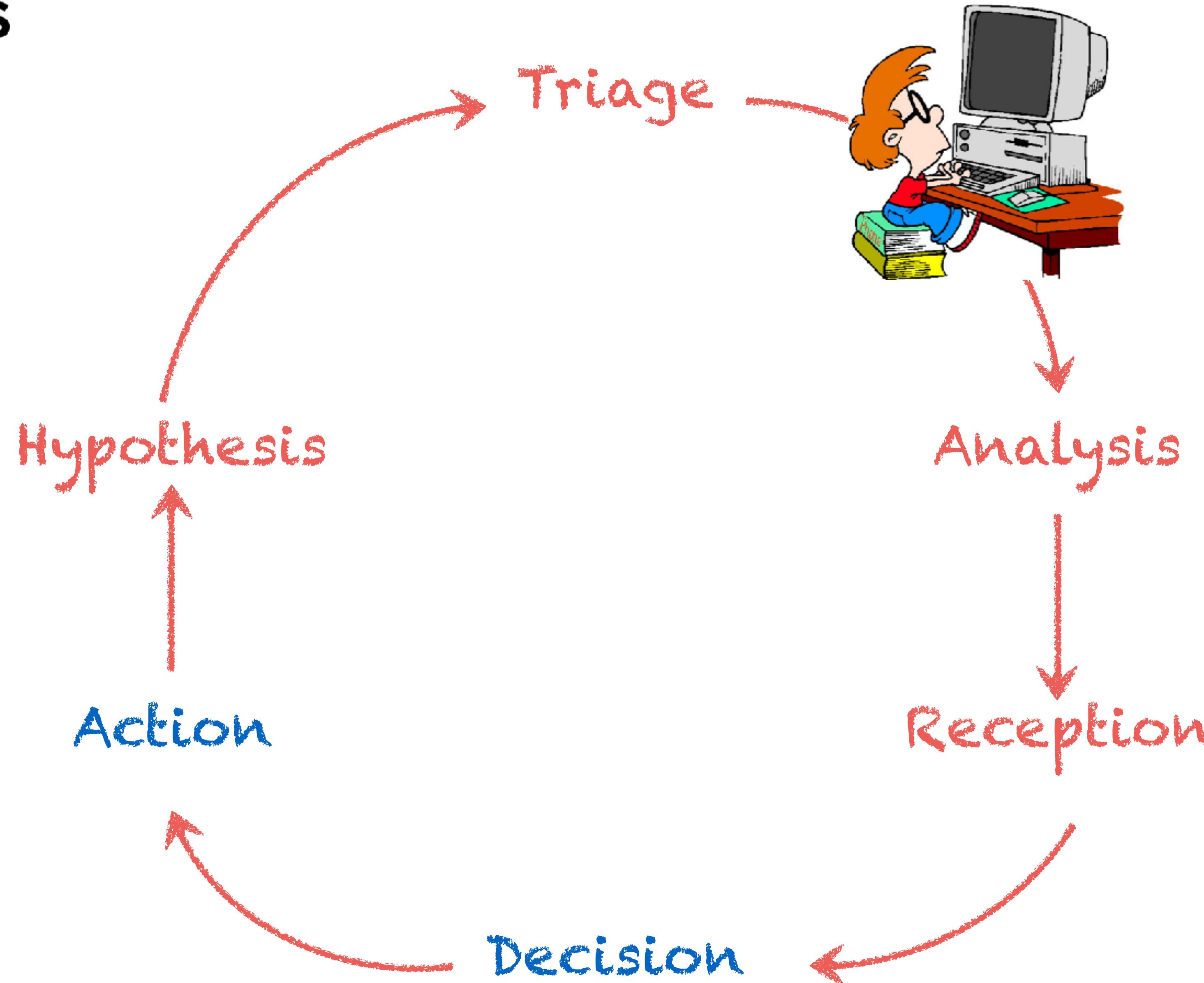


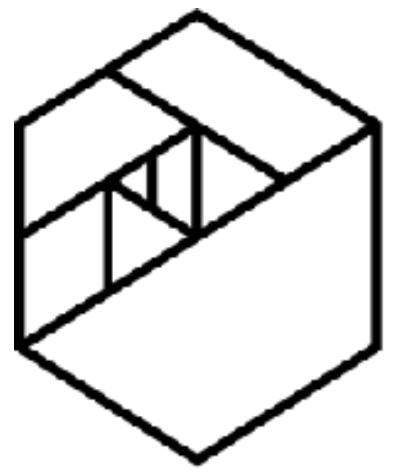
METIS





METIS





**METIS**

# **Thoughts for Data Science Success**

# Data is a Strategic Asset... not a cost



# **Align Projects to Corporate Strategy**

# Align Projects to Corporate Strategy



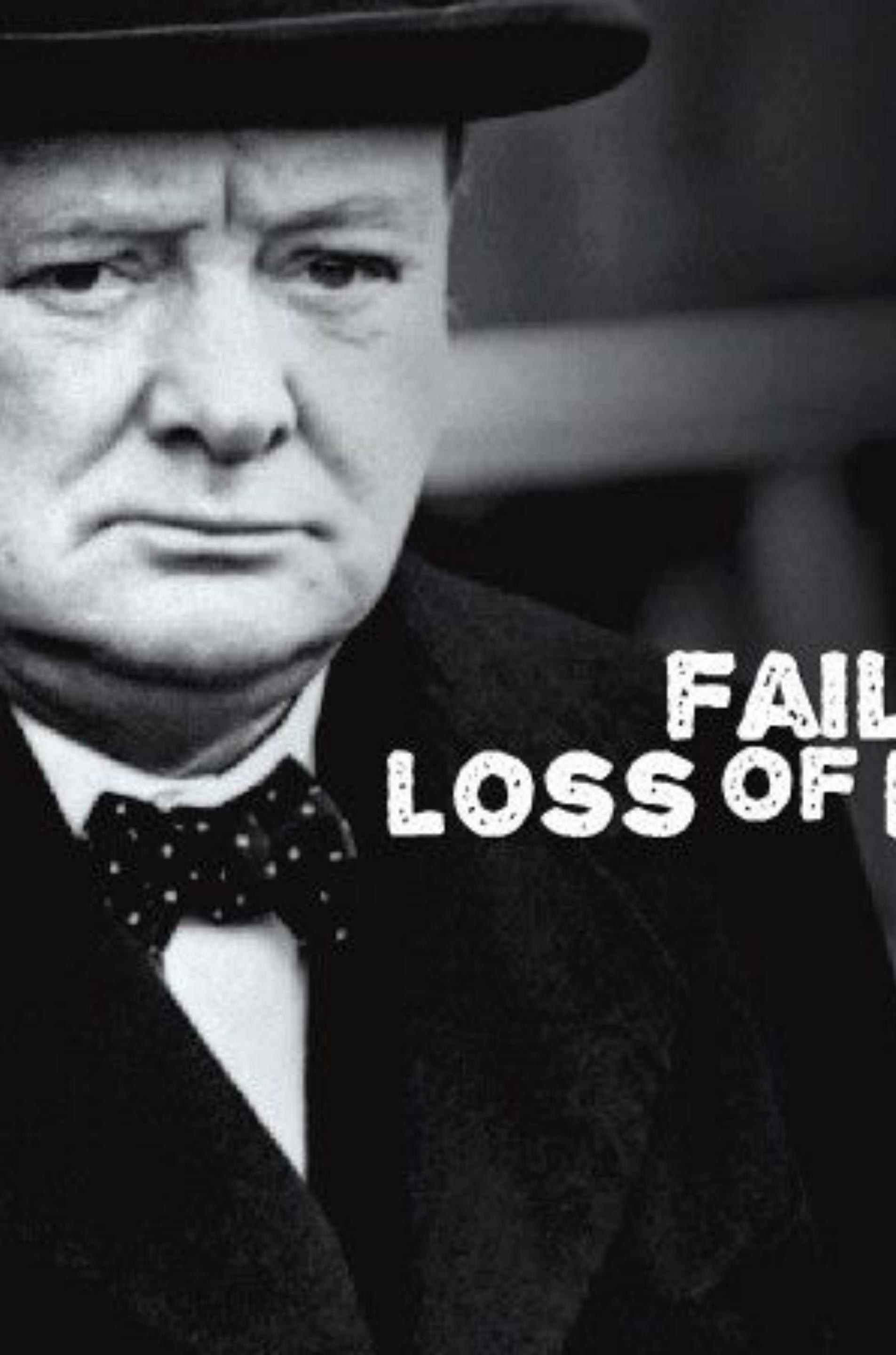
Your:  
Time  
Money  
Job?

**Build the right team for Data Initiatives**



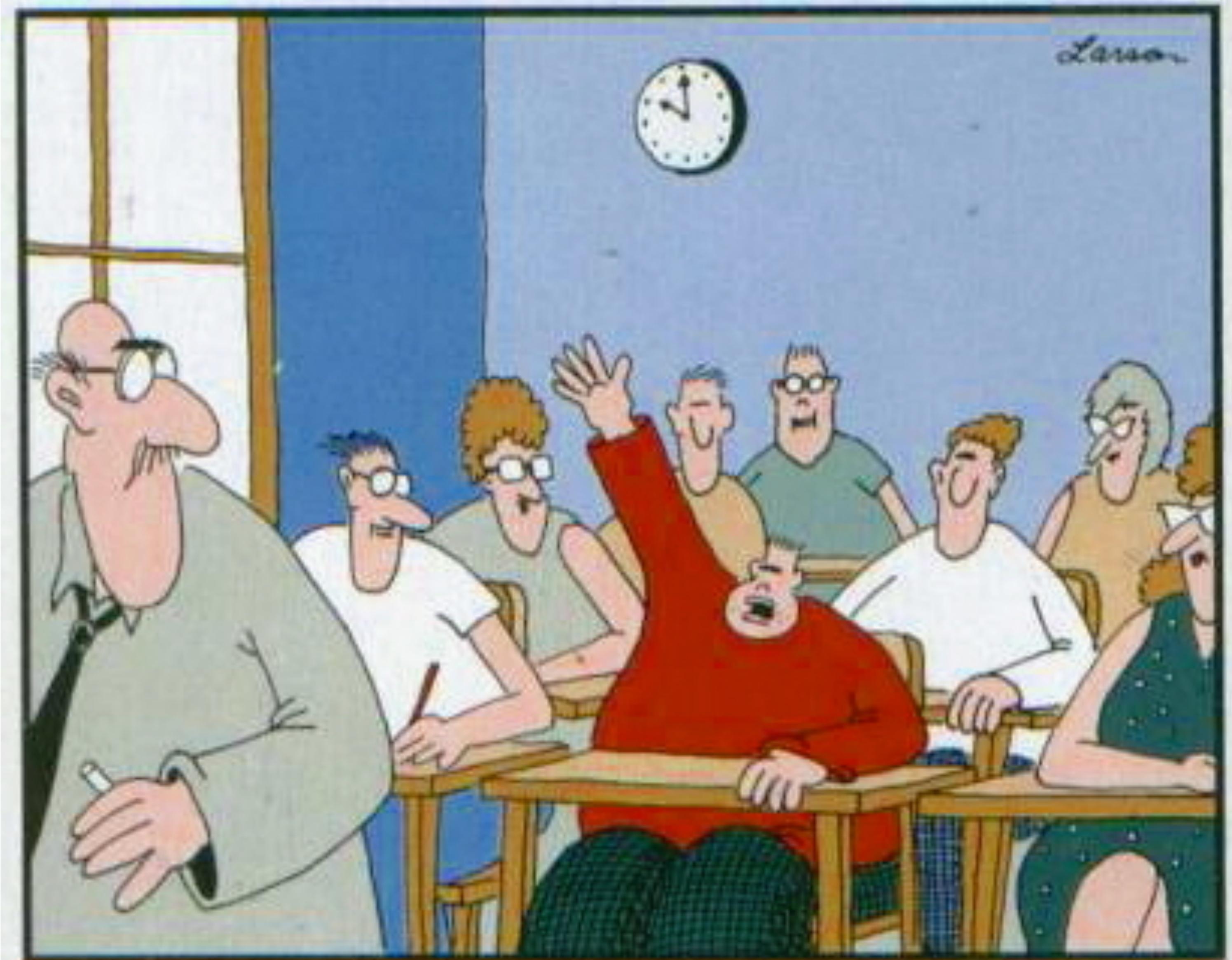
Prioritize building appropriate data platform



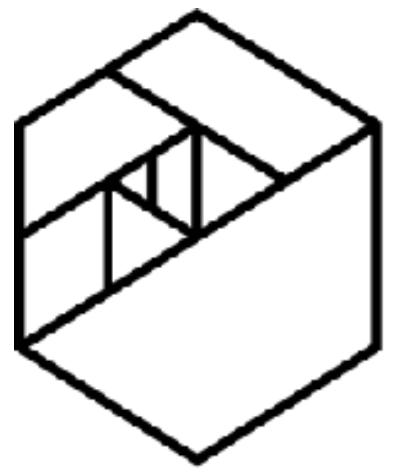


**"SUCCESS  
CONSISTS OF  
GOING FROM  
FAILURE TO  
FAILURE WITHOUT  
LOSS OF ENTHUSIASM."**

Winston Churchill

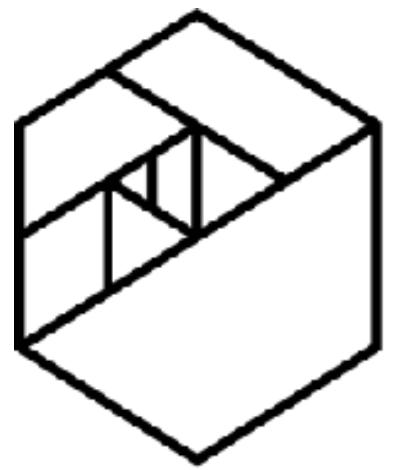


**"Mr. Osborne, may I be excused?  
My brain is full."**



METIS

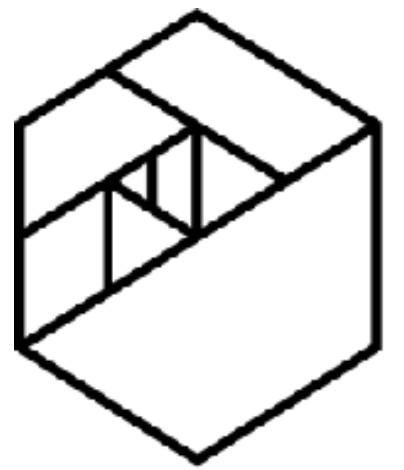
# Building a Data Science Team



METIS

# Building a Data Science Team

Programmer + Statistician + SME = DS Team?

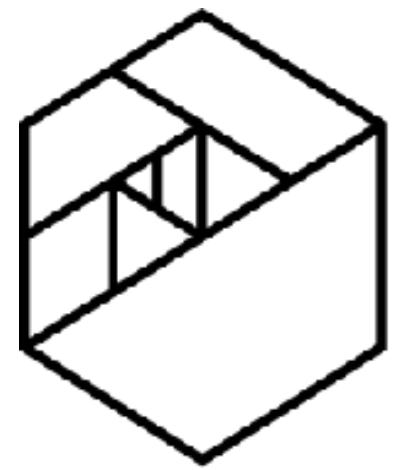


METIS

# Building a Data Science Team

Programmer + Statistician + SME = DS Team?

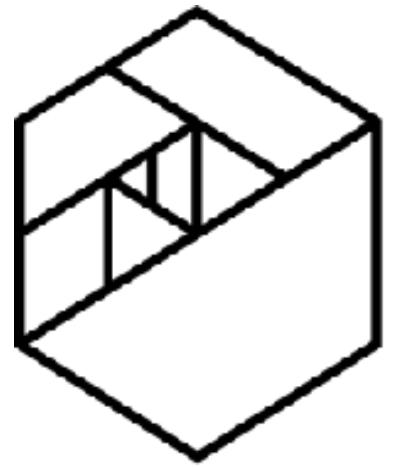
Sort of...



METIS

# Building a Data Science Team

- Data Ambassador
- Data Scientist
- Data Storyteller
- Data Engineer

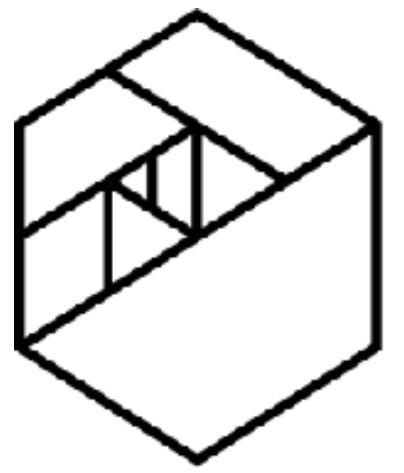


METIS

By the end of the class  
you will be able to:

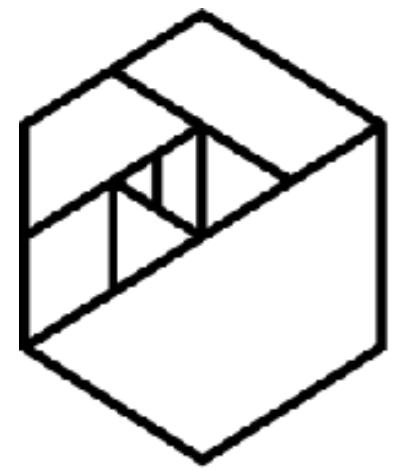
- Quickly and effectively prepare data for analysis
- Apply machine learning techniques to enhance security





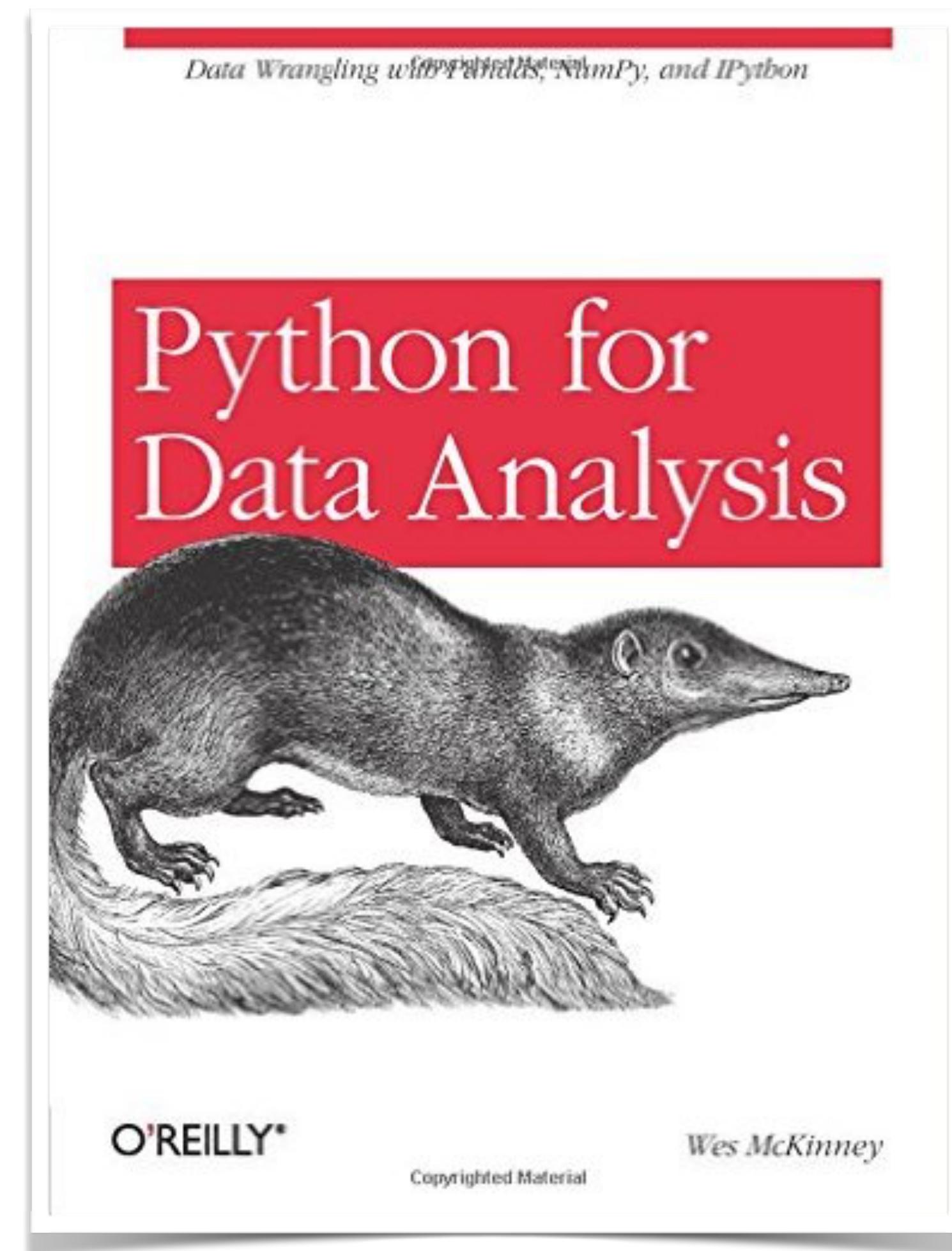
METIS

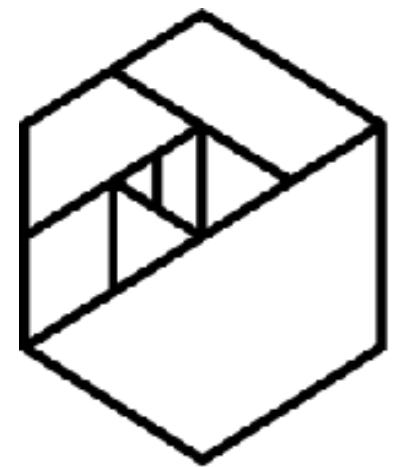




METIS

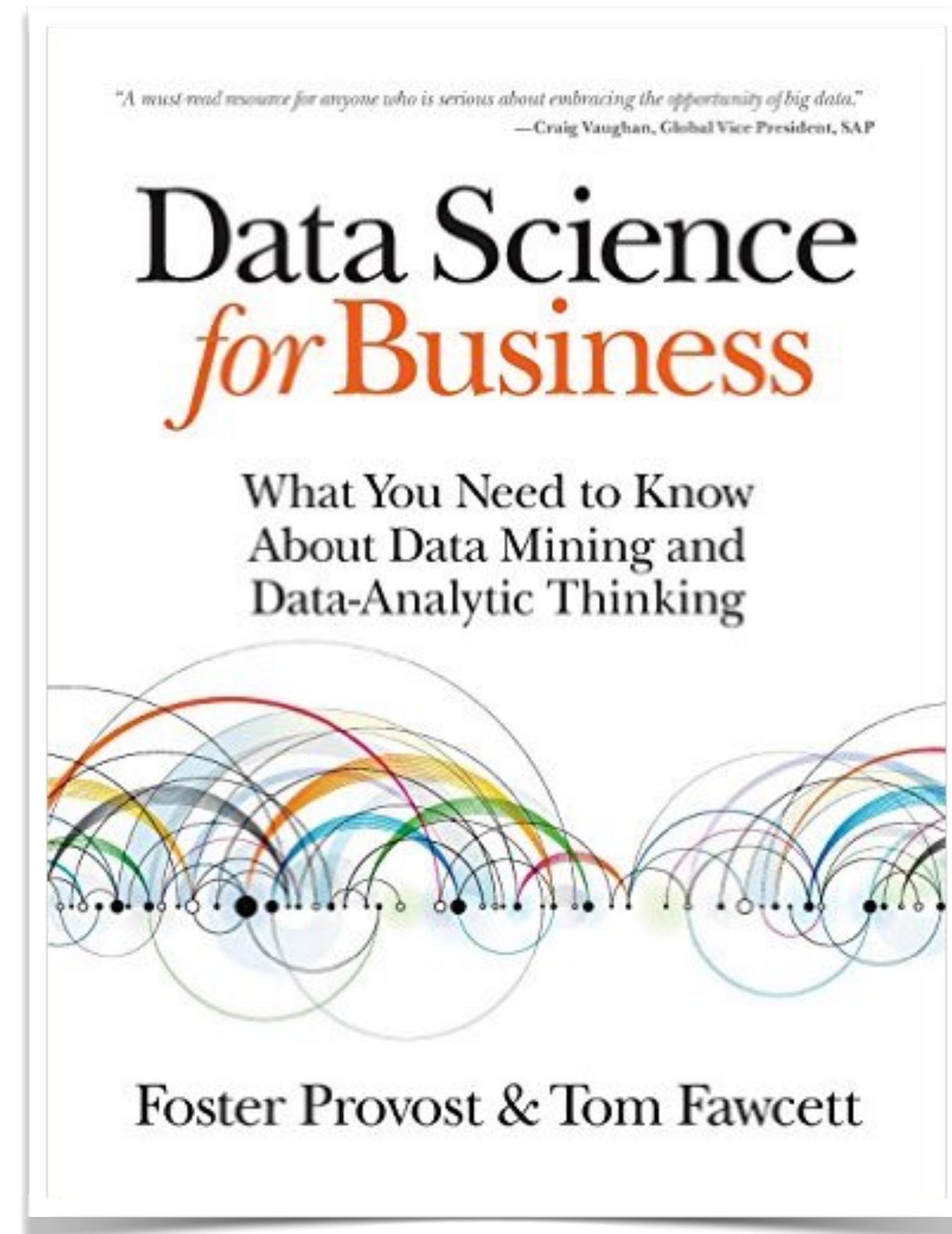
# Recommended Reading

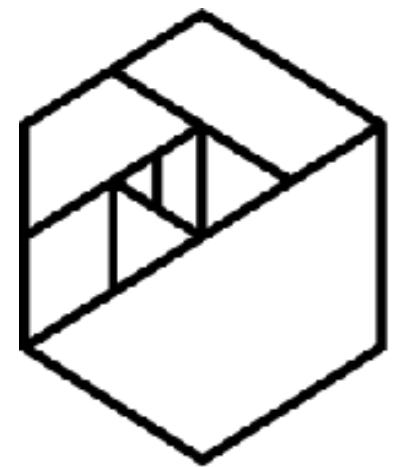




METIS

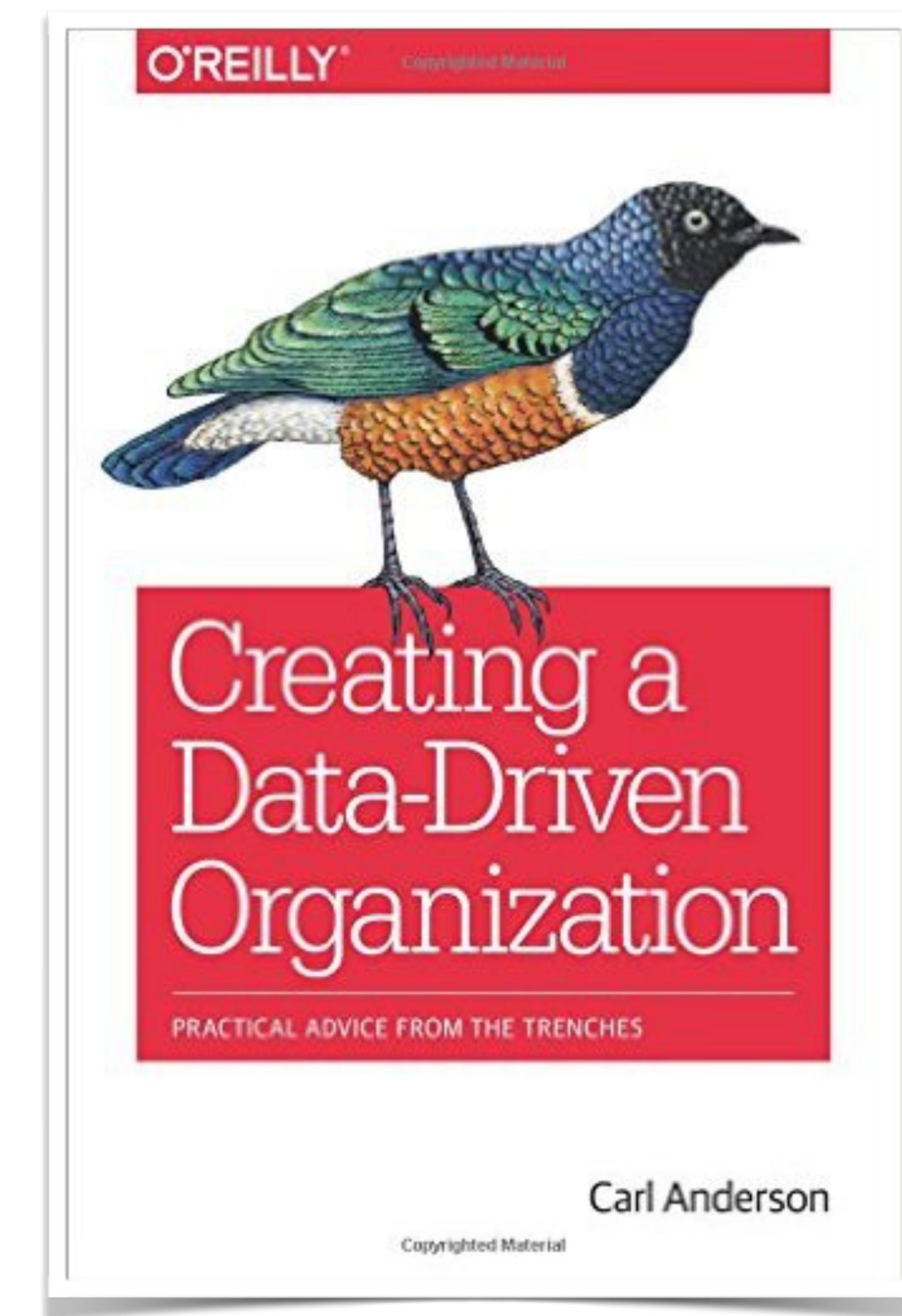
# Recommended Reading

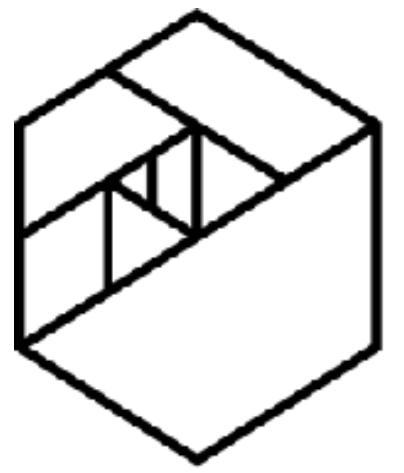




METIS

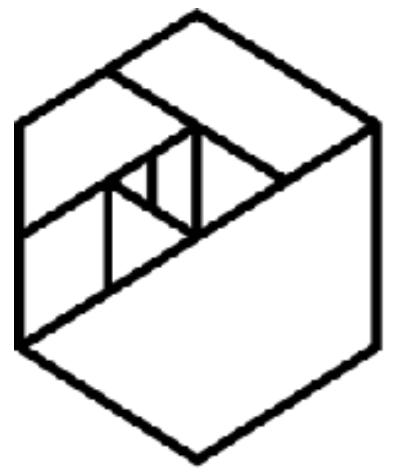
# Recommended Reading





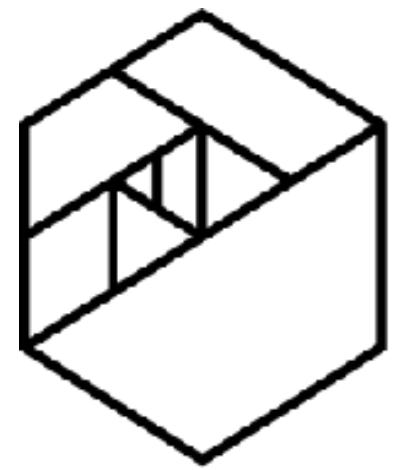
**METIS**

# Questions?



**METIS**

# Jupyter Notebook



METIS

# Jupyter Notebook

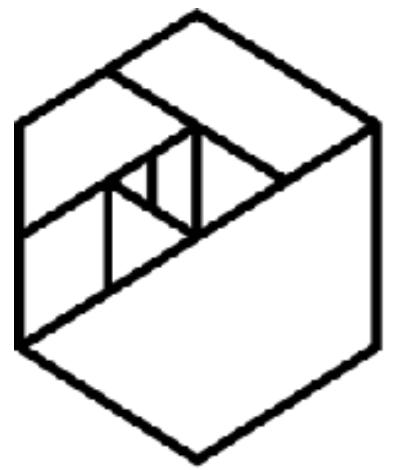
---

jupyter 01\_intro\_to\_ipython\_notebook Last Checkpoint: 8 hours ago (autosaved)  [Logout](#)

File Edit View Insert Cell Kernel Help Not Trusted Python [default] O

Markdown

---

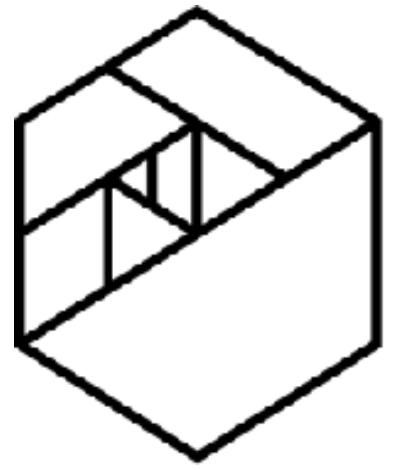


METIS

# Jupyter Notebook

```
In [1]: print("Welcome to Intro to Data Science!")
```

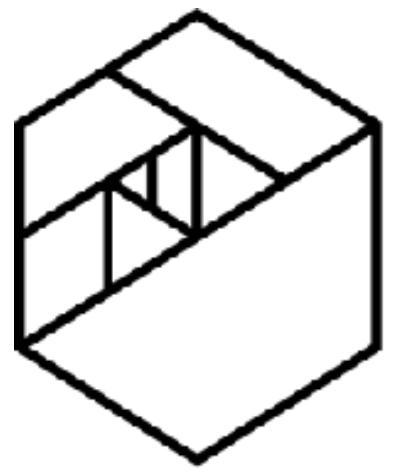
```
Welcome to Intro to Data Science!
```



METIS

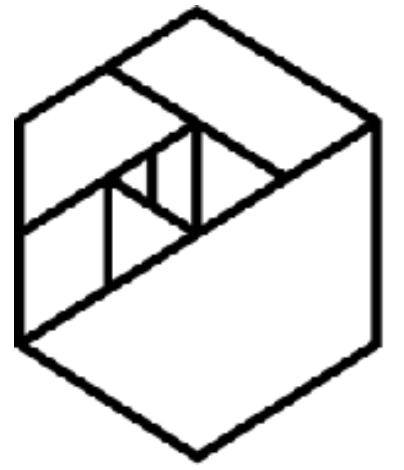
# Exercise

- Please take 10 minutes to acclimate yourself to the Jupyter Notebook



**METIS**

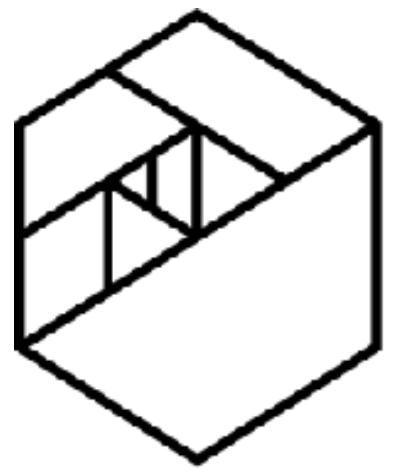
# Version Control



METIS

# Version Control

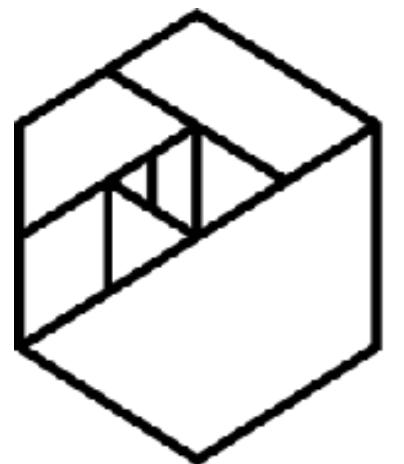
## What is it?



METIS

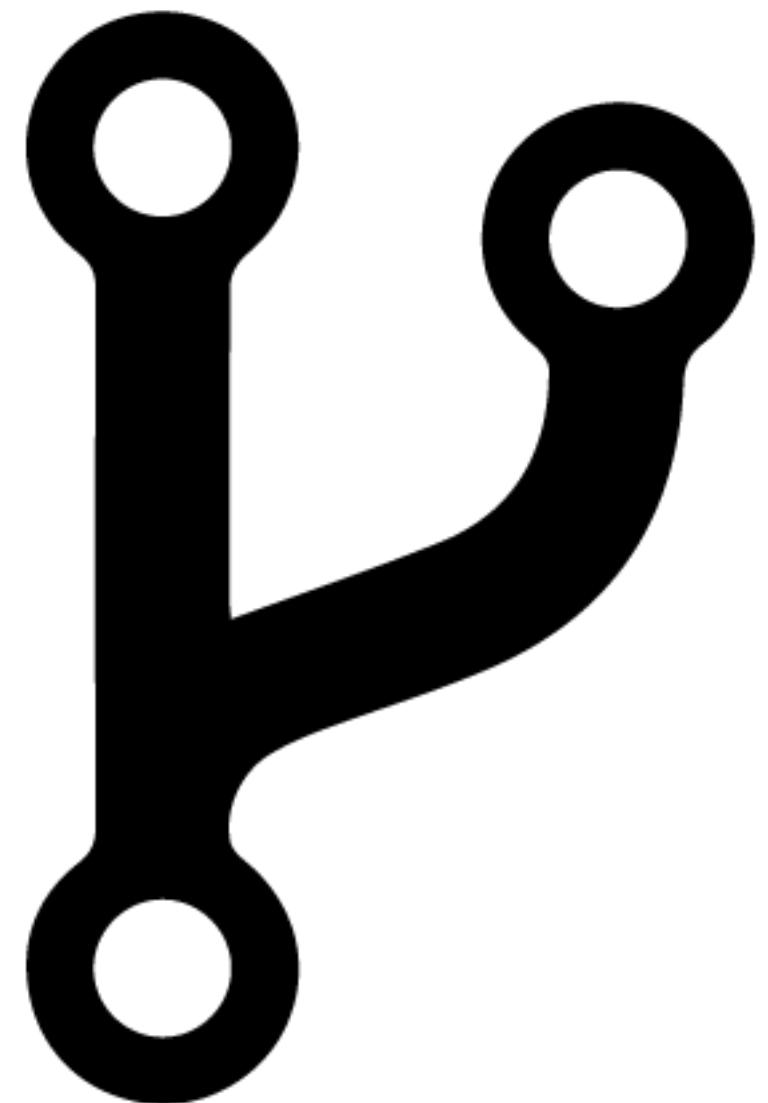
# Version Control

[https://github.com/thisismetis/dc17\\_ids1](https://github.com/thisismetis/dc17_ids1)



METIS

# Version Control

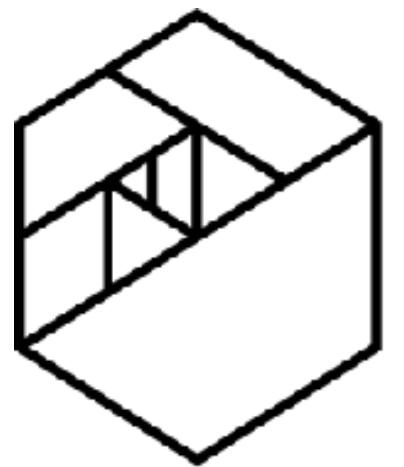


Forking a Repository

[https://github.com>thisismetis/dc17\\_ids1](https://github.com>thisismetis/dc17_ids1)

Unwatch 3   Star 0   Fork 0

Code Issues 0 Pull requests 0 Projects 0 Wiki Settings Insights



METIS

# Version Control

## Cloning a Repository

The screenshot shows a GitHub repository page for a user named "thisismetis". At the top, there are three buttons: "Create new file", "Upload files", and "Find file". To the right of these is a green button labeled "Clone or download ▾". A large black arrow points from the text above to this button. Below the buttons is a dropdown menu with two options: "Clone with HTTPS" (with a question mark icon) and "Use SSH". Underneath the dropdown, a URL is displayed: [https://github.com>thisismetis/dc17\\_ids1.g](https://github.com>thisismetis/dc17_ids1.g). To the right of the URL is a clipboard icon. At the bottom of the dropdown, there are two more buttons: "Open in Desktop" and "Download ZIP". Below the dropdown, the text "8 hours ago" is visible.

Create new file   Upload files   Find file   Clone or download ▾

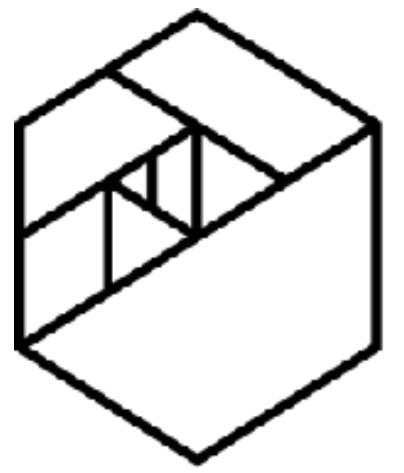
Clone with HTTPS ⓘ   Use SSH

Use Git or checkout with SVN using the web URL.

[https://github.com>thisismetis/dc17\\_ids1.g](https://github.com>thisismetis/dc17_ids1.g)

Open in Desktop   Download ZIP

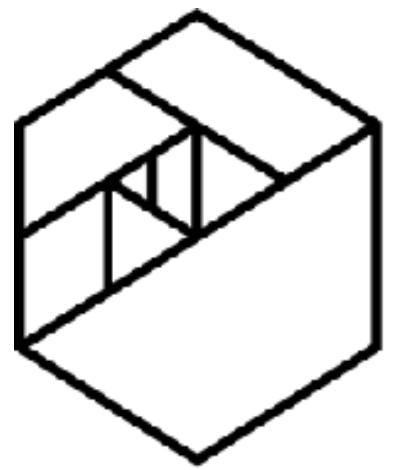
8 hours ago



METIS

# Version Control

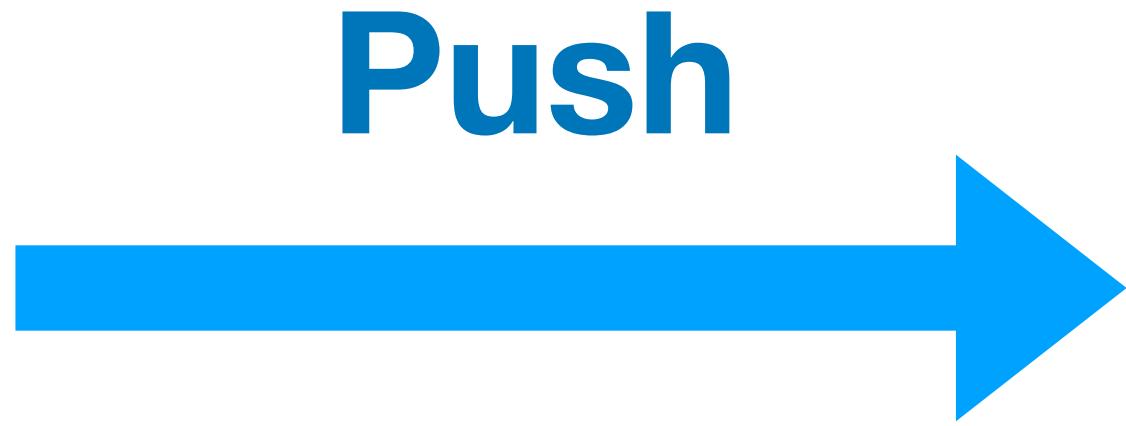
```
git clone https://github.com/thisismetis/dc17_ids1
```



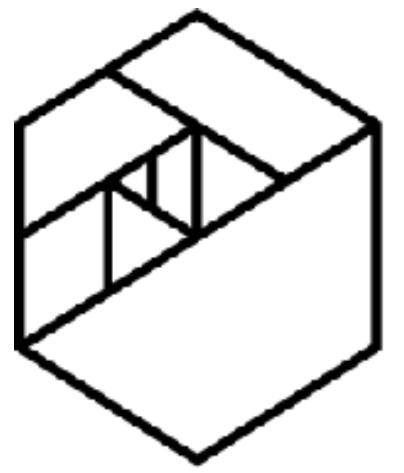
**METIS**



Local



Remote



**METIS**



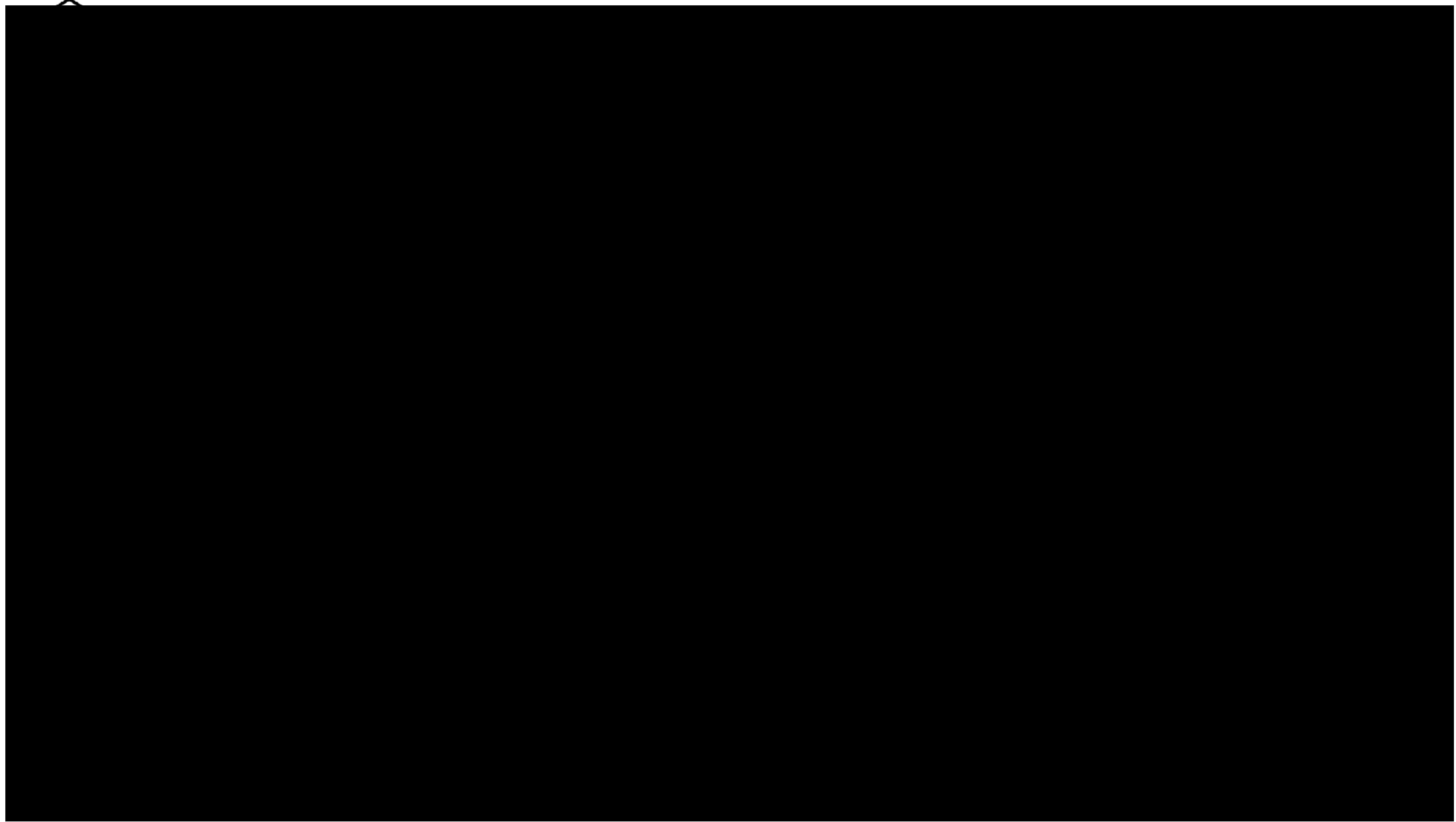
Local

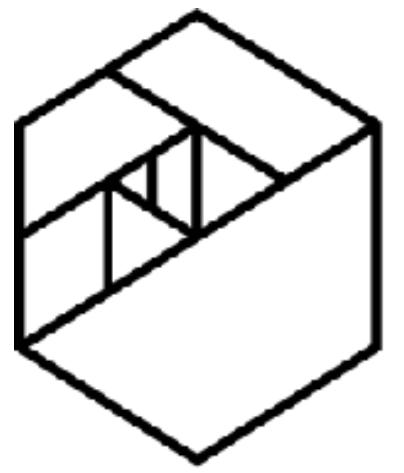


**Pull**



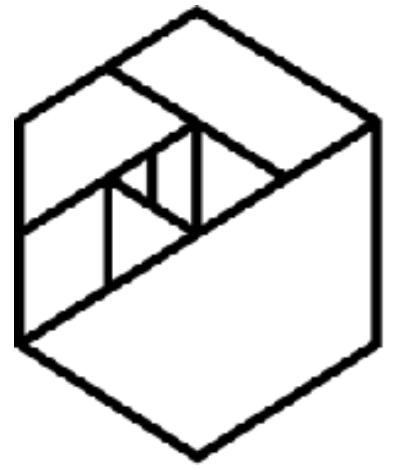
Remote





**METIS**

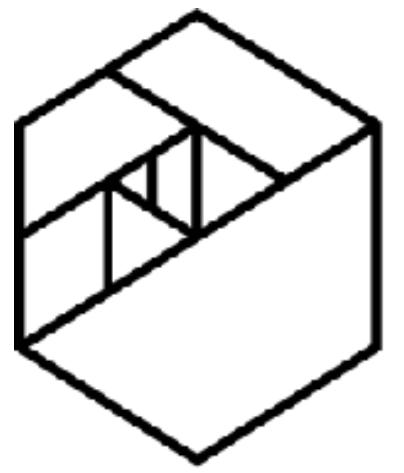
**Before we start coding...**



METIS

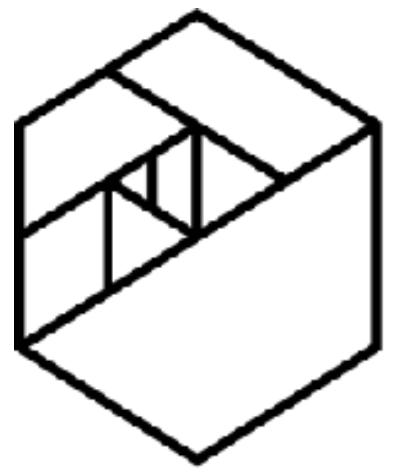
# A Simple Problem

- Write down the steps necessary to calculate the minimum number of coins necessary to give change. IE:  $.62 = 2$  Quarters, 1 Dime, 2 Pennies



**METIS**

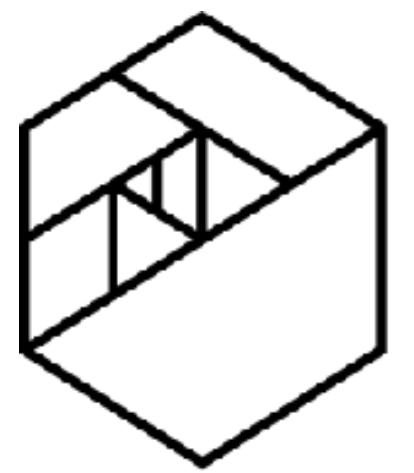
# Atoms of Programming



METIS

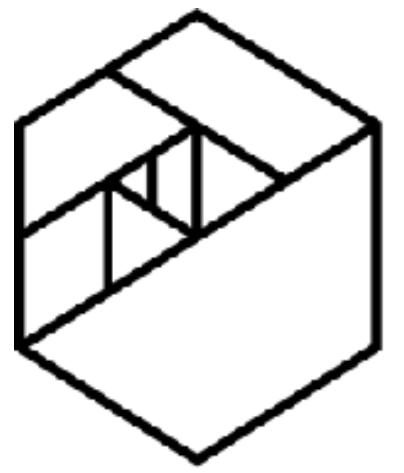
# Atoms of Programming

- Variables
- Math (Arithmetic)
- Assignment
- Conditional Logic (If/Else)
- Iteration
- Collections
- Functions
- Input/Output



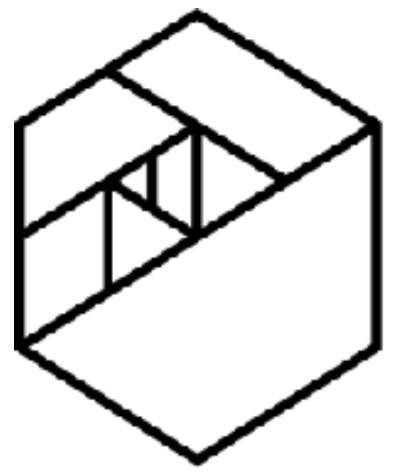
**METIS**





**METIS**

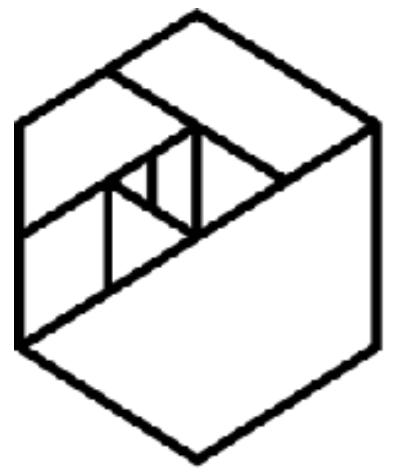
# Atoms of Programming



METIS

# Python Data Types

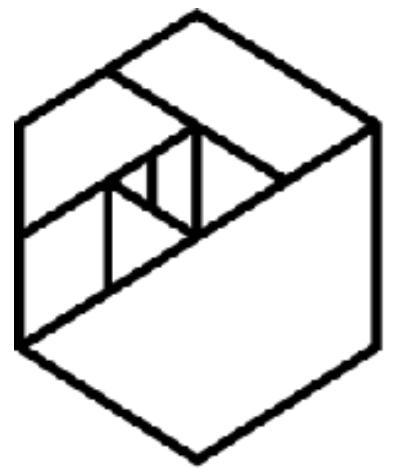
- Integer
- Floating Point
- String
- Boolean



METIS

# Python Data Types

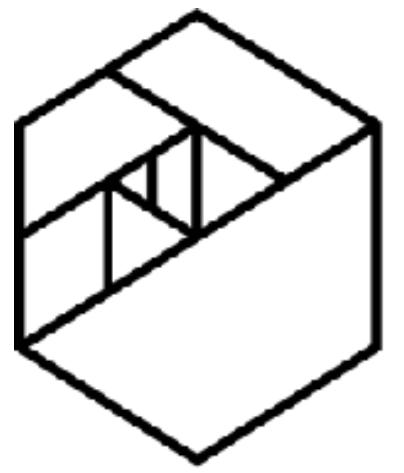
x = 5



METIS

# Python Data Types

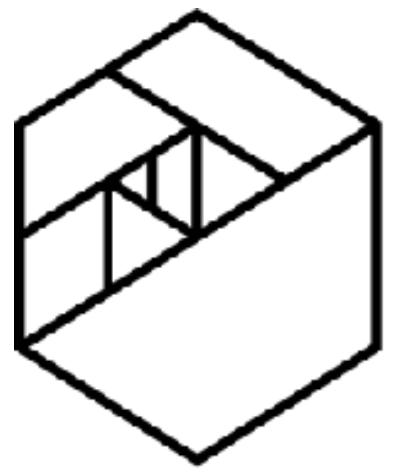
<code>type( 5 )</code>	<code>int</code>
<code>type( 5.0 )</code>	<code>float</code>
<code>type(True)</code>	<code>bool</code>
<code>type("test")</code>	<code>str</code>



METIS

# Python Data Types

```
x = 5
type(x)          int
x = 5.0
type(x)          float
```



METIS

# Python Data Types

```
x = 5
```

```
type(x)
```

```
type(float(x))
```

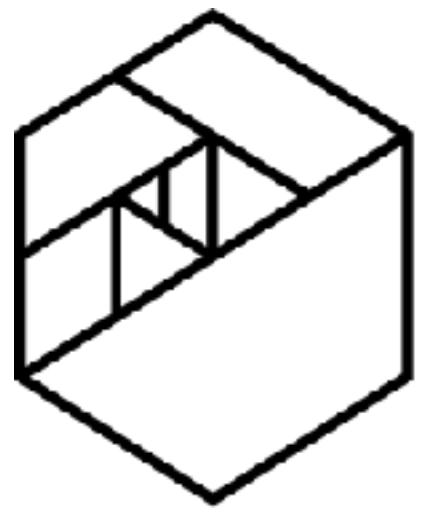
int

float

```
y = "test"
```

```
float(y)
```

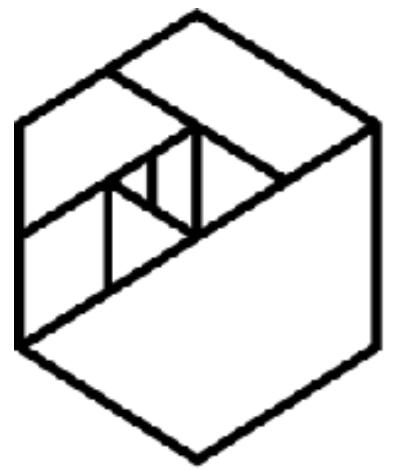
#Error



METIS

# Quick Exercise

Declare a few variables and convert them to different data types  
Which conversions don't work?



METIS

# Comments

Comments are code that does not get executed.

Python has two styles: multiline and single

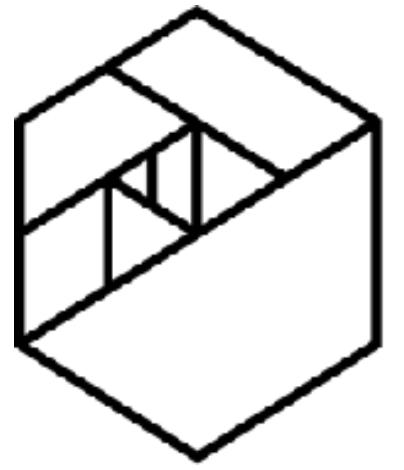
```
'''
```

**This is a multi-line comment**

**More comment**

```
'''
```

**#Single line comment**

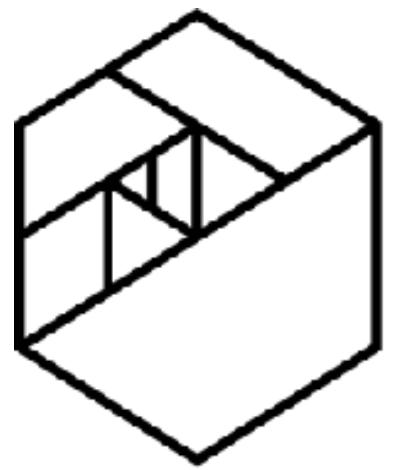


METIS

# Getting Help

Python has a REALLY useful function called **help()** which gets you the documentation for variables or functions.

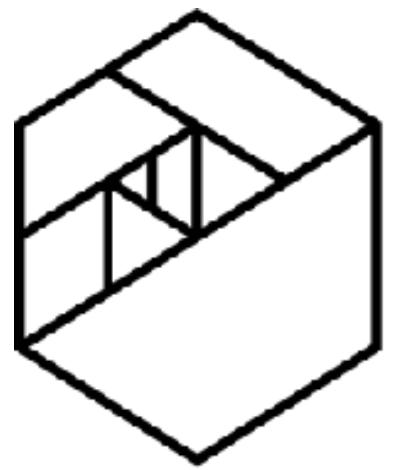
Give it a try.



# Lists

- Python and many other languages have a concept of variables that contain many variables. In Python, these are referred to as a list.
- They are indexed from zero

Index	Value
0	“First”
1	“Second”
2	“Third”
3	“Fourth”



METIS

# Lists

```
#Creating a list  
myList = ["first", "second", "third",  
"fourth"]
```

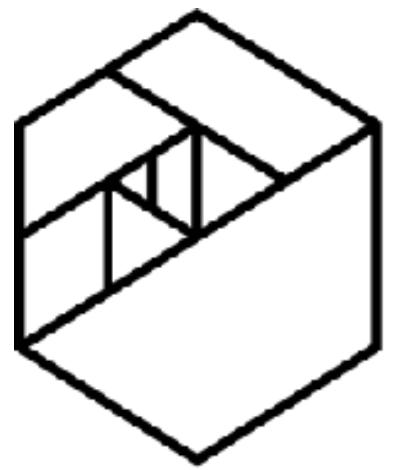
```
#Accessing a list item  
print( myList[0] )  
>> First
```

```
#Adding to a list  
myList.append("fifth")
```

```
#Getting list length  
len( myList)  
>> 5
```

What other list methods are available?

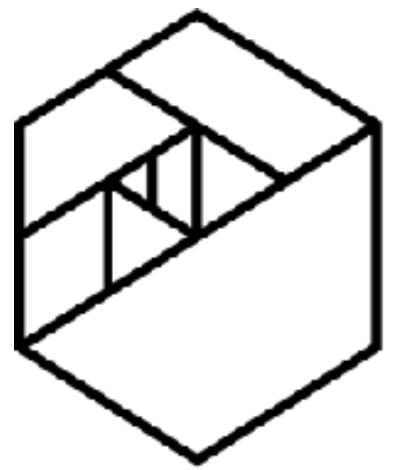
Index	Value
0	"First"
1	"Second"
2	"Third"
3	"Fourth"



METIS

# List Slicing

```
myList[ startIndex : endIndex : stepSize ]
```



METIS

# List Slicing

```
#Get every other item
```

```
myList = ["first", "second", "third",  
"fourth"]
```

```
myList[1::2]
```

```
>> ["Second", "Fourth"]
```

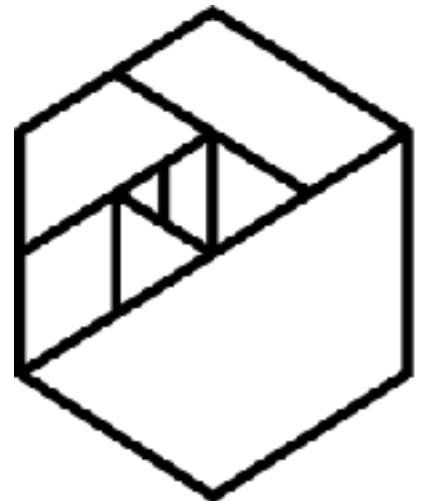
```
#Reverse the list
```

```
myList[::-1]
```

```
>>
```

```
["fifth", "fourth", "third", "second", "firs  
t"]
```

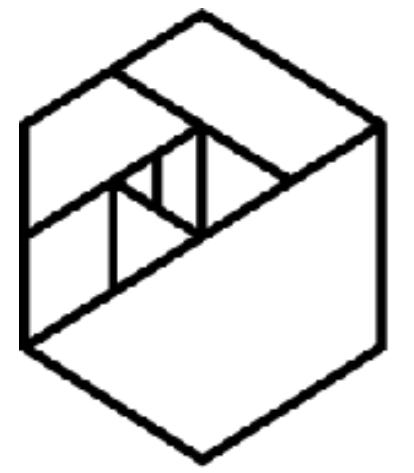
Index	Value
0	"First"
1	"Second"
2	"Third"
3	"Fourth"



METIS

# Quick Exercise

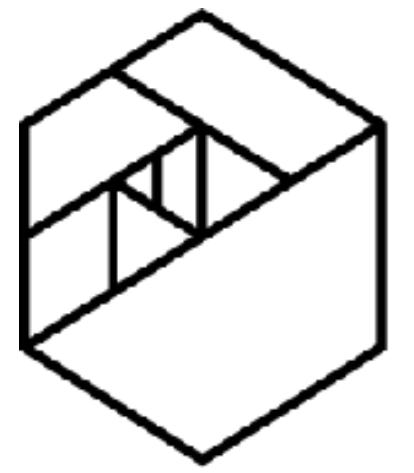
1. Create a list of 10 numbers
2. Reverse it
3. Sort it
4. Add two more numbers to it
5. Create a new list with every other item from the original list



METIS

# Strings

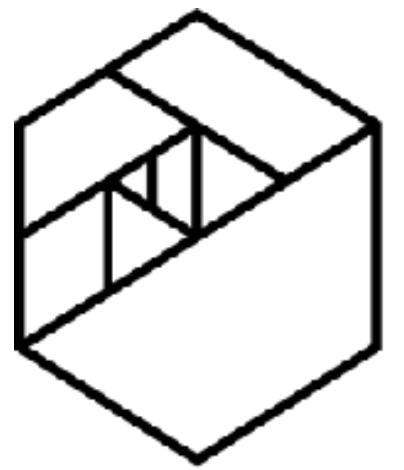
```
myName = "Charles S Givre"
```



METIS

# Strings

```
myName = "Charles S Givre"
```



METIS

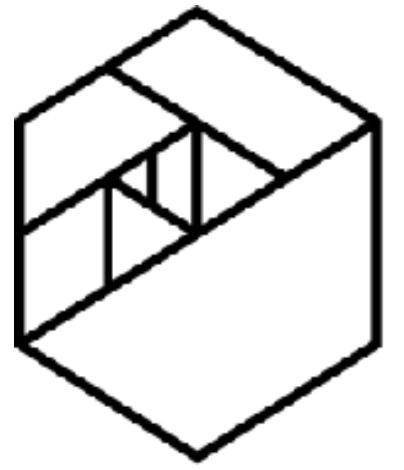
# Strings

```
myName = "Charles S Givre"
```

```
#You can slice strings  
firstName = myName[0:7]
```

```
len(firstName)  
>> 7
```

```
print(firstName)  
>> Charles
```



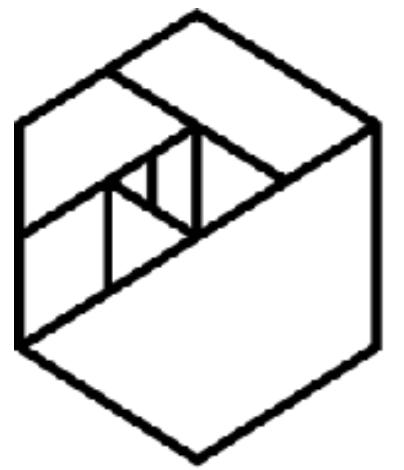
METIS

# Strings

```
myName = "Charles S Givre"
```

```
#You can split strings  
nameParts = myNames.split()
```

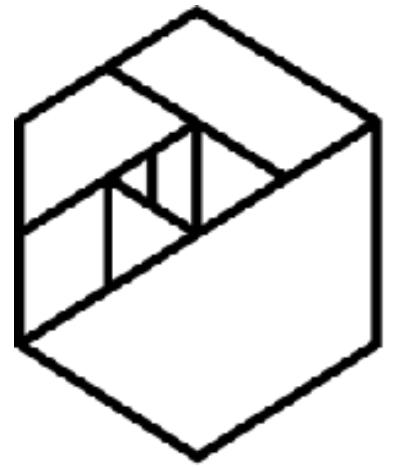
Try this...



METIS

# Strings

- There are MANY useful functions that are associated with the String object including:
  - `upper()`
  - `lower()`
  - `capitalize()`
  - `isalpha()`
  - and more...

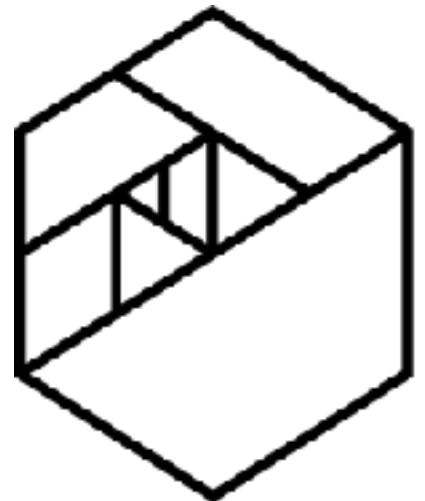


# METIS The Range Function

The range function produces a sequential list of integers.

```
range( start, end, step )
```

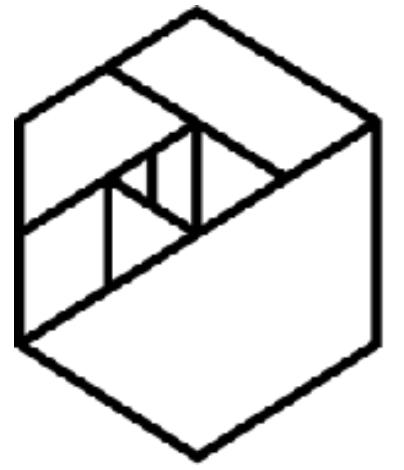
The end is NOT inclusive and the step size is optional.



METIS

# Exercise

- Create a variable `my_new_list` and set it to contain "dude" and the string "55"
- Create a new variable `dude55` that is the concatenation of "dude" and "55"
- Create a variable `my_int` that is the int representation of "55"
- Create a new string called `my_substring` that is the 3rd through 5th characters of `dude55`
- Create a list called `my_range` that is all the multiples of 3 from 3-26



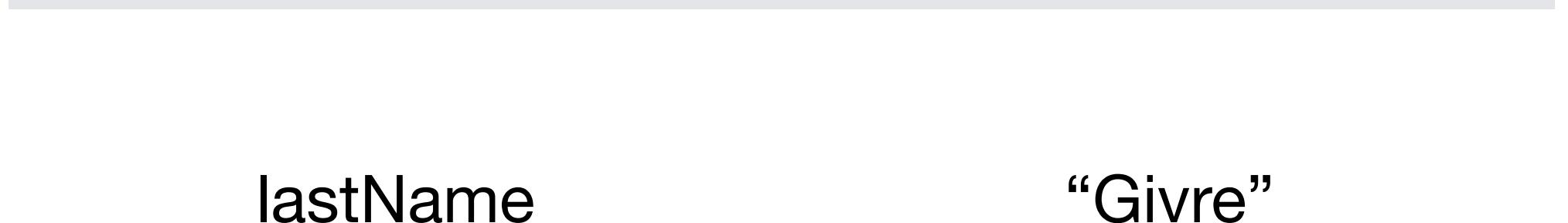
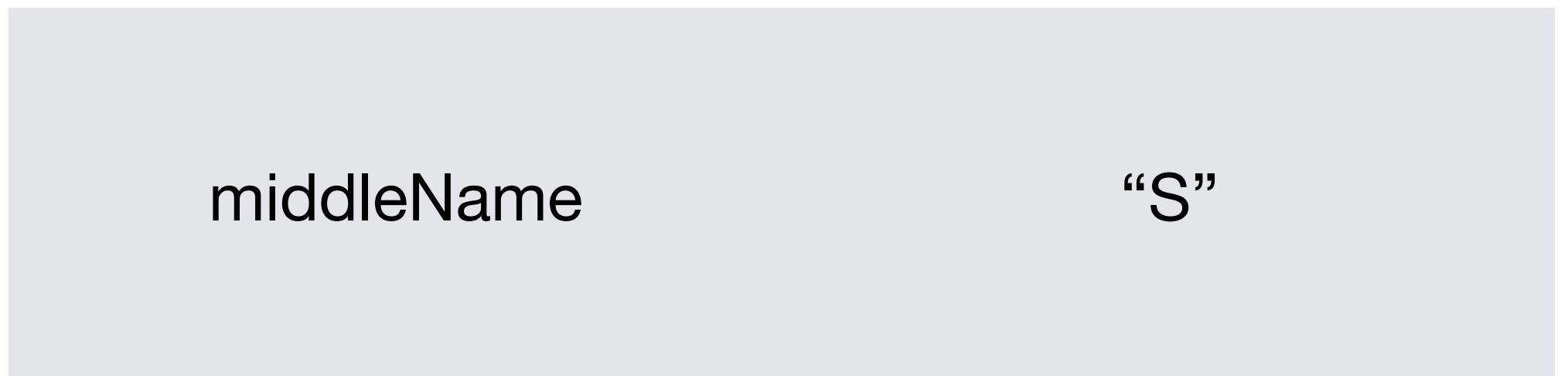
METIS

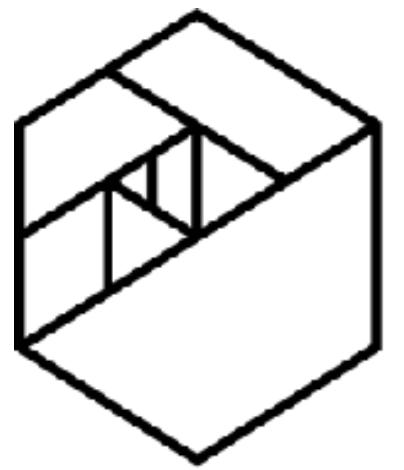
# Dictionaries

- Dictionaries are similar to lists, however they are indexed by key instead of by position



- Keys must be unique
- Dictionaries have no order





METIS

# Dictionaries

```
#create a dictionary
```

```
record = {"firstName": "Charles",
          "middleName": "S",
          "lastName": "Givre"
        }
```

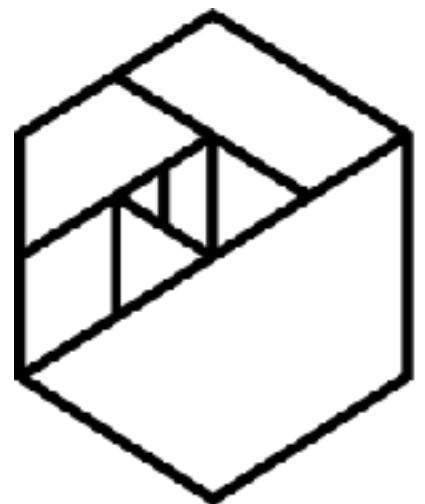
```
#Accessing a dictionary item
```

```
print( record['firstName'] )  
>> Charles
```

```
#Adding a dictionary item
```

```
record['Salutation'] = "Mr."
```

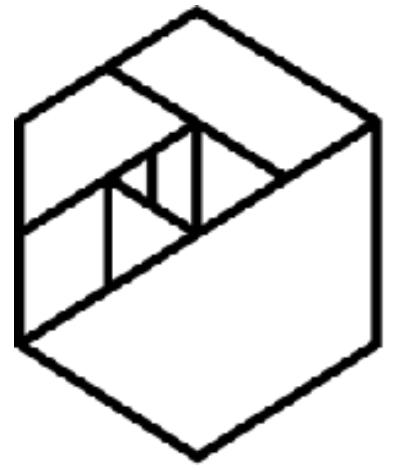
Index	Value
firstName	"Charles"
middleName	"S"
lastName	"Givre"



METIS

# Exercise

- From the worksheet, print the name of the best borough (in the dictionary).
- Create a new key-value pair for new jersey (give it any value you like)
- Fix the boroughs stored in the other boroughs object so that they're both upper case

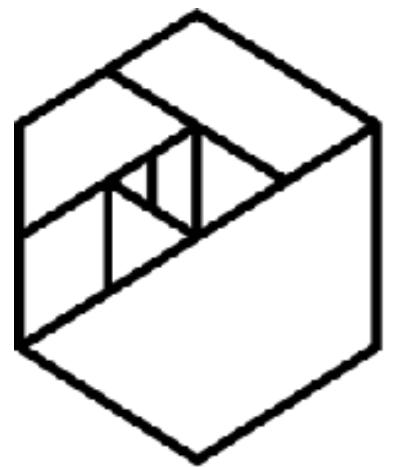


METIS

# Functions

Functions are reusable blocks of code.

```
def isEven( x ):  
    return x % 2 == 0
```



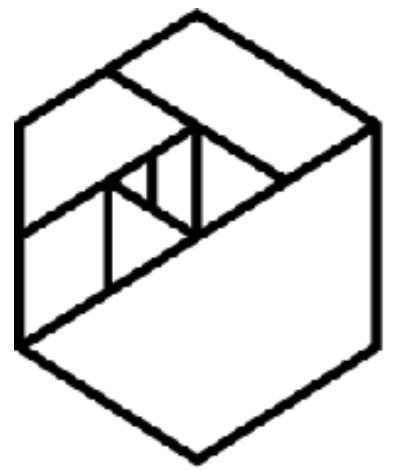
METIS

# Functions

```
def reverseArray( myArray ):  
    return myArray[ ::-1 ]
```

```
def addSalesTax( price, taxRate ):  
    return price + (price * taxRate)
```

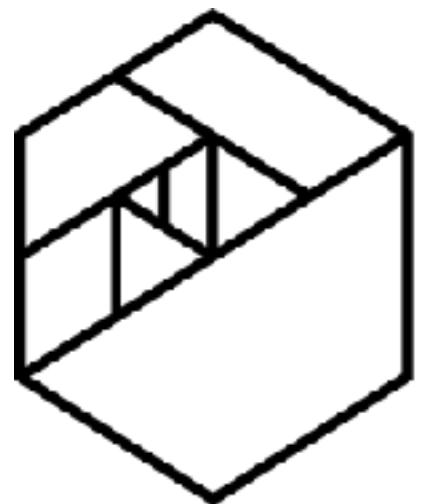
```
#calling a function  
addSalesTax( 100.0, 0.06 )  
>> 106.0
```



METIS

# Functions

```
def calc_default_add(x, y, op="add"):  
    if op == 'add':  
        return x + y  
    elif op == 'subtract':  
        return x - y  
    else:  
        print('Valid operations: add, subtract')
```

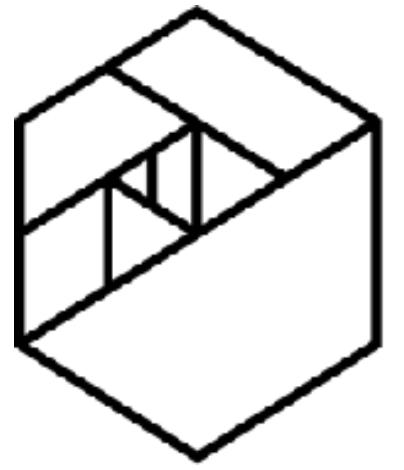


METIS

# Exercise

Write two functions:

- One called `compute_pay` that takes two parameters (`hours` and `rate`), and returns the total pay.
- One called `get_hours_worked` that takes two parameters (`total_pay` and `rate`) and returns the total hours worked.

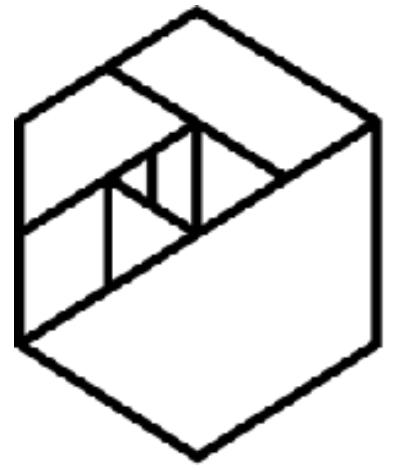


METIS

# Iteration

One of the programming atoms is the concept of iteration. In Python, the basic loop is the `for` loop.

```
for i in range(0,5):  
    print(i)
```

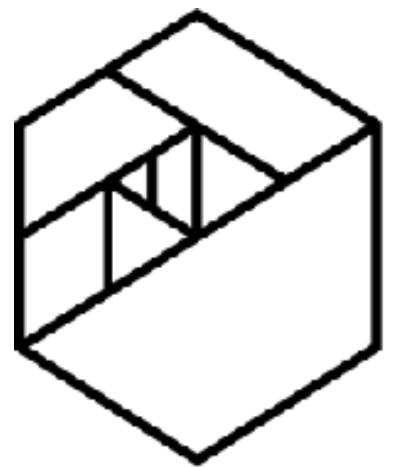


METIS

# Iteration

## What does this do?

```
names = [ 'bob' , 'steve' , 'sally' , 'sue' ]  
for name in names:  
    print(name)
```



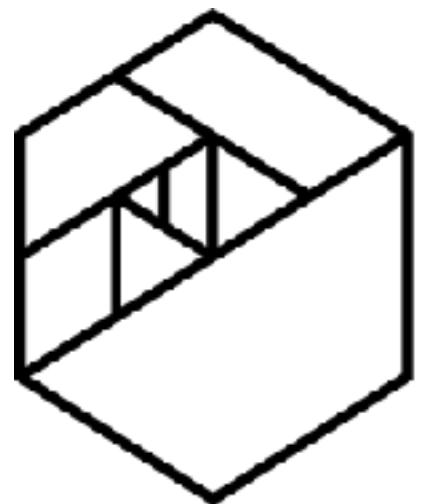
METIS

# Iteration: List Comprehensions

A list comprehension is shorthand for a simple loop. Often they are used for mathematical calculations.

```
squares = [ x * x for x in range(1,11) ]
```

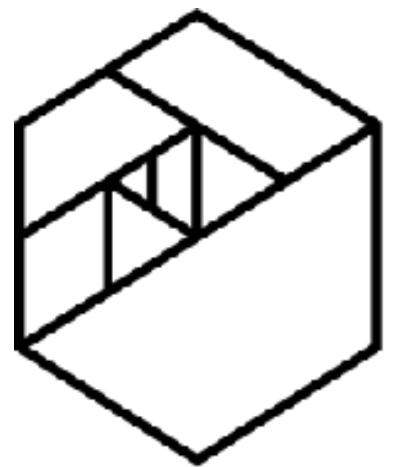
```
squares = []
for x in range(1,11):
    squares.append( x * x )
```



METIS

# Exercise

- Given `words = ['yo', 'hello', 'awesome']` write a list comprehension that returns `["YO", "HELLO", "AWESOME"]`
- Given `word = "fancy"` write a list comprehension that returns `['F', 'A', 'N', 'C', 'Y']`
- Write a function called `awesome_sauce` that prints the numbers from 1 to 100. However, for multiples of 2 it should print 'awesome' instead of the number, and for multiples of 7 it should print 'sauce' instead of the number, and for numbers which are multiples of both 2 and 7 it must print 'awesome sauce!'.



METIS

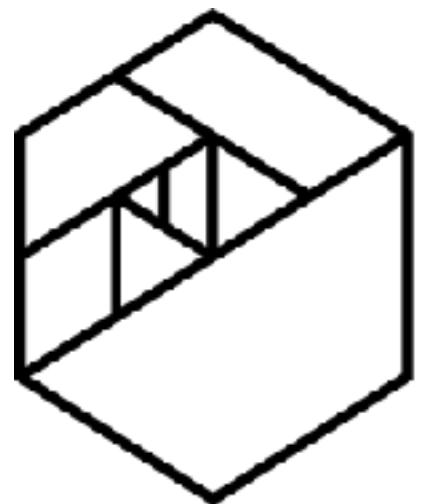
# Loading Data

To load data from the internet, we will have to use some of Python's amazing modules.

```
import csv
import requests

with open('<file>', 'r') as f:
    vertebral_data = [row for row in csv.reader(f)]

#print the first 5 elements in vertebral_data
for line in vertebral_data[:5]:
    print(line)
```



METIS

# Exercise

- Split every item in `iris_data` on the commas
- Split every item in `vertebral_data` on the spaces
- Get only the numeric entries in each item in `iris_data`