Intro^2 to Data Analysis and Machine Learning

Data 602, Lecture 1

Instructor: Chris McGraw

Outline for today

- Introductions
- Course logistics
- Grading
- Tentative schedule
- Resources
- Begin Data analysis using pandas

DATA 602

Introduction to Data Analysis and Machine Learning

- Anyone not here for DATA602 should unceremoniously creep to the exits (or silently log-off).

- This course provides a broad introduction to the practical side of **machine-learning** and **data analysis**.

- This course examines the **end-to-end processing pipeline** for **extracting and identifying useful features** that best represent data, a few of the most important **machine algorithms**, and **evaluating their performance** for modeling data. Topics covered include decision trees, logistic regression, linear discriminant analysis, linear and non-linear regression, basic functions, support vector machines, neural networks, Bayesian networks, bias/variance theory, ensemble methods, clustering, evaluation methodologies, and experiment design.

Instructor Information

Chris McGraw

- **Education:**
  - B.S., Statistics (UMBC)
  - M.A., Applied Economics (Johns Hopkins University)
- **Professional:**
  - Analytics focused roles since 2008 (~13 years)
  - Analytics/data science at not-for-profit, public sector, Fortune 500 companies
- **Personal**:
  - Two (2) young kids (so I don’t sleep much)
  - Nine (9) 300 games in bowling
  - Watched every movie in the Marvel cinematic universe at least 5 times

Introduce yourself!

Name, why are you in the program, what are you most excited for in this class, maybe an interesting thing about you?

Course Logistics

- **Meeting time**: Tuesdays, 7:10pm - 9:40pm
- **Office hours**: Thursdays, 7:00pm - 9:00pm, or by appointment (both will be via WebEx)
- **Course dates**: 8/31/2021 - 12/13/2021
- **Location**:
  - 8/31 and 9/7 virtual via WebEx
  - 9/14 - 12/13 Information Technology 231
- **Book**: Raschka and Mirjalili; Python Machine Learning, 3rd Edition
- **Official software**: Python (download Anaconda or use Google Colab)
- **Instructor**: Chris McGraw (he/him)
  - chrism3@umbc.edu (preferred - I will reply quickly)
  - 410/688/4680 (SMS or call, emergencies only please)
  - If you text, let me know who you are :)

Grading

Homework, mid-terms, and the final project

- **Homework**: 70% (~ 10 Homework assignments)
- **Final project**: 30%
- **Grading**
  - A: 90-100%
  - B: 80-89%
  - C: 70-79%
  - D: 60%-69%
  - F: <60%

Academic Integrity Policy

TL;DR - No cheating!

By enrolling in this course, each student assumes the responsibilities of an active participant in UMBC's scholarly community in which everyone's academic work and behavior are held to the highest standards of honesty. Cheating, fabrication, plagiarism, and helping others to commit these acts are all forms of academic dishonesty, and they are wrong. Academic misconduct could result in disciplinary action that may include, but is not limited to, failure, suspension or dismissal. Refer to the UMBC policy at: http://catalog.umbc.edu/content.php?catoid=14&navoid=718#academic-integrity

To take an easy example, would you want to be operated on by a doctor who cheated his way through medical school? Or would you feel comfortable on a bridge designed by an engineer who cheated her way through engineering school. Would you trust your tax return to an accountant who copied his exam answers from his neighbor?

Bill Taylor, “Academic Integrity: A Letter to My Students”

Academic Integrity

What I (the instructor) am required to do

- Show up to class
- Be on time
- Do my best to answer your questions
- Acknowledge if I don’t have the answer and that I’ll research it by next class
- Not allow others to ridicule you or your ideas - there are no stupid questions and its better to ask if you are unsure of even a minor detail
- I’ll let you know what’s my opinion versus a substantiated fact

Integrity Related to Assignments/Coding

It ain’t worth it

- No sharing of code
  - General question, e.g., ‘how can I aggregate multiple fields in a data frame’, can be asked via the discussion board or researched from online sources
- The risk for cheating is highly asymmetric:
  - Pro: pass the course
  - Cons: fail the course, suspended, fail the next course, fired from employer after exposed, …
- I’ll be reviewing your code
  - I can extract the code (I’m a data scientist)
  - I can compare the code (I’m a data scientist)
  - I don’t want to have to compare the code
  - If you borrow code from a source, cite it - that’s what people in academia and professional do

Blackboard

- Syllabus is posted on Blackboard (and we will review now)
- Lectures will be posted under Course Materials
- Assignments will be posted under Assignments (and I'll discuss in-class)
  - Be mindful of the due date (generally midnight day of next class)
- Discussions
  - Feel free to use these, especially if you have questions or have found resources that may benefit the entire class
  - I'll monitor and respond as needed

Tentative schedule

Things could shift depending on pacing and interest

08/31	**1**	Introduction	
09/07	**2**	Data analysis and pandas I	HW Due
09/14	**3**	Data analysis and pandas II	HW Due
09/21	**4**	Machine learning tour	HW Due
09/28	**5**	Designing your experiment	HW Due
10/05	**6**	Feature engineering	HW Due
10/12	**7**	Regression I	HW Due
10/19	**8**	Regression II	HW Due

10/26	**9**	Evaluating classification models	HW Due
11/02	**10**	Classification models I	HW Due
11/09	**11**	Classification models II	HW Due
11/16	**12**	Classification models III	
11/23	**13**	Unsupervised learning	
11/30	**14**	Dimensionality reductions and other considerations	
12/07	**15**	Project Presentations	Project Due

Each class we'll start by reviewing homework from the week before

My Goals

At the end of the course you should have learned…

- The typical data science lifecycle
- Best practices for exploratory data analysis
- Be able to map a data science problem to potential solutions
- Be able to explain what types of information a machine can learn
- Be conversant in various machine learning models and be able to construct them independently
- Be able to evaluate performance of machine learning models
- Be able to provide constructive feedback to peers on their data science work

Helpful Resources

- https://scikit-learn.org/stable/
- https://numpy.org
- https://pandas.pydata.org
- https://matplotlib.org
- https://seaborn.pydata.org
- http://themlbook.com
- https://sebastianraschka.com
- https://scikit-learn.org/stable/tutorial/machine_learning_map/index.html
- https://rviews.rstudio.com
- https://www.kdnuggets.com

Resources for Datasets

You'll need a dataset for the project (think early)

- https://www.kaggle.com
- http://lib.stat.cmu.edu/datasets/
- https://github.com/rfordatascience/tidytuesday
- https://scikit-learn.org/stable/modules/classes.html?highlight=datasets#module-sklearn.datasets
- https://github.com/UMBC-Data-Science/DATA602Datasets/

Tools

- We will be using Python and its associated data science libraries, primarily:
  - pandas (a lot)
  - numpy (a lot)
  - scikit-learn (a lot)
  - matplotlib (a lot)
  - sqlite3 (a little)
- Recommend installing Anaconda
  - https://www.anaconda.com/products/individual
  - Includes all the major packages that we'll be using
  - Also includes Jupyter Notebook, which will be the supported IDE
- Google Colab is optional (Google's version of Jupiter Notebooks)

Tips for being successful

- How do you get to Carnegie Hall
- Ask questions, talk with your classmates
- Don’t fall in love with a particular method
- Simple generally beats complicated
- Create lots of graphs
- Features over models
- This is as much an art than a science
- Dissect problems to their lowest level, then try to solve
- Think how an executive would use the information you produce
- Presentation and story are usually more important than what model you used

Questions?

Course, logistics, anything at all?

Data Analysis

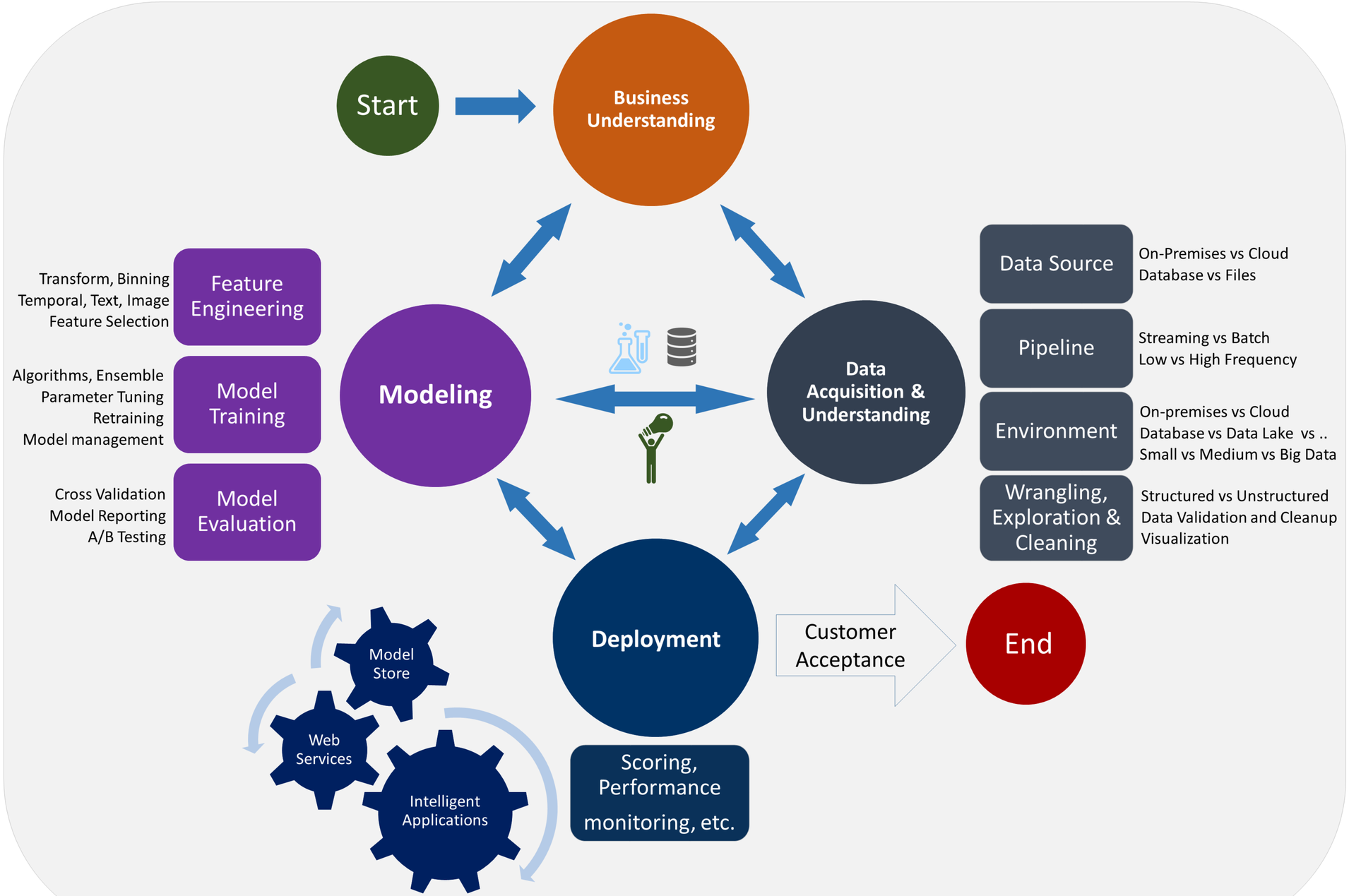
The less glamorous, but most critical step in the process

Why bother?

Can’t we just artificial intelligence it?

- Understand the problem and frame a solution.
- Understand what data is available, assess its quality, and determine potential features.
- Determine how to fully utilize the data we have available.
  - What data don’t we have that would be useful? Can we create it? Collect it?
- Determine and utilize appropriate models.
- Evaluate performance and communicate results.
  - Convey statistical performance vs. business performance

Data Science Lifecycle



Source: https://docs.microsoft.com/en-us/azure/architecture/data-science-process/overview, Accessed 7/18/21

“Machine learning is a thing-labeler, essentially.”

Cassie Kozyrhov, Chief Decision Scientist at Google, Inc.


RED HAT
SUMMIT

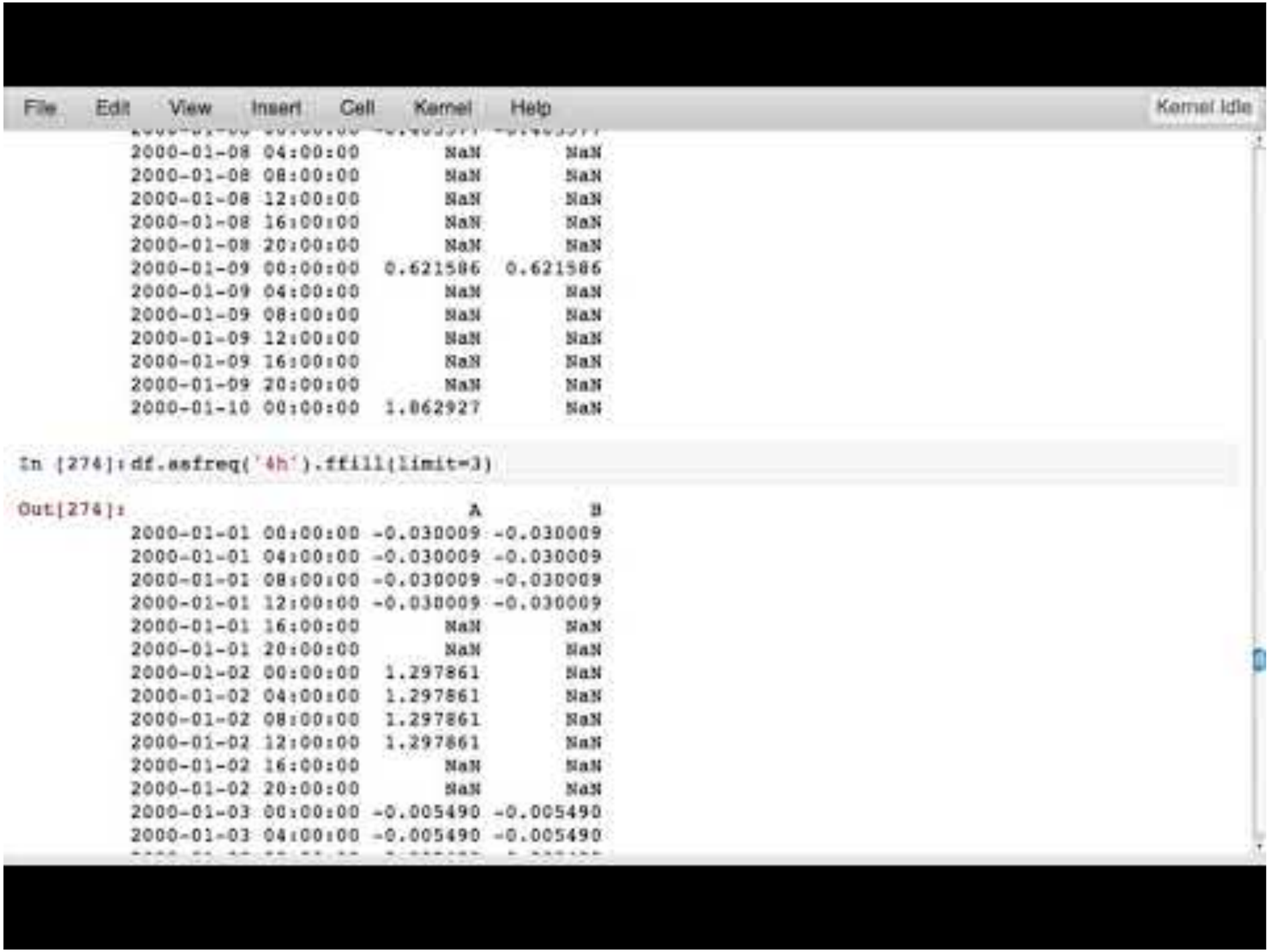
TL;DR Cassie's talk

Ignoring the Google Cloud and RedHat pitches…

- The toolbox is full - no need to reinvent the regression wheel
  - You don't necessarily need to know how to code a neural net from scratch
  - Even with a fully stocked machine learning kitchen, you need a chef that knows what they are doing
  - You should know when to use it, what can go wrong, and best practices
- Creativity and solving problems is the real challenge
- And there is one tiny detail is always neglected about machine learning…

Real data is messy.

Luckily, there is pandas


File Edit View Insert Cell Kernel Help
Kernel Idle
2000-01-08 04:00:00 NaN NaN
2000-01-08 08:00:00 NaN NaN
2000-01-08 12:00:00 NaN NaN
2000-01-08 16:00:00 NaN NaN
2000-01-08 20:00:00 NaN NaN
2000-01-09 00:00:00 0.621586 0.621586
2000-01-09 04:00:00 NaN NaN
2000-01-09 08:00:00 NaN NaN
2000-01-09 12:00:00 NaN NaN
2000-01-09 16:00:00 NaN NaN
2000-01-09 20:00:00 NaN NaN
2000-01-10 00:00:00 1.062927 NaN
In [274]: df.asfreq('4h').ffill(limit=3)
Out[274]:
A B
2000-01-01 00:00:00 -0.030009 -0.030009
2000-01-01 04:00:00 -0.030009 -0.030009
2000-01-01 08:00:00 -0.030009 -0.030009
2000-01-01 12:00:00 -0.030009 -0.030009
2000-01-01 16:00:00 NaN NaN
2000-01-01 20:00:00 NaN NaN
2000-01-02 00:00:00 1.297861 NaN
2000-01-02 04:00:00 1.297861 NaN
2000-01-02 08:00:00 1.297861 NaN
2000-01-02 12:00:00 1.297861 NaN
2000-01-02 16:00:00 NaN NaN
2000-01-02 20:00:00 NaN NaN
2000-01-03 00:00:00 -0.005490 -0.005490
2000-01-03 04:00:00 -0.005490 -0.005490

Let’s get our Python muscles warmed up

Lab activity - feel free to follow along, but I’ll post this to Blackboard

- You are an investor in seniors housing / nursing home properties
- You know through market research the majority of residents in these properties are 85 or older
- Your boss wants a report on recommendations to jump into the market
- The US Census Bureau prepares population projections
  - It contains all the information you need, but the format isn’t conducive for analytics

Population projections data

See population-projections notebook

- Retrieve data from: https://www2.census.gov/programs-surveys/popproj/datasets/2017/2017-popproj/np2017_d1_mid.csv
- File layout: https://www2.census.gov/programs-surveys/popproj/technical-documentation/file-layouts/2017/np2017_d1.pdf

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4860 entries, 0 to 4859
Columns: 106 entries, SEX to POP_100
dtypes: int64(106)
memory usage: 3.9 MB
```

Homework - Due 9/7/21 by midnight

WHO Tuberculosis Data

- Borrowed from the “tidyr” package in R
- Documentation: https://tidyr.tidyverse.org/reference/who.html
- “who” is a very wide dataset that needs cleaning before it’s really usable for analysis
- See Blackboard > Assignments
- Between the pandas video and the Census data lab you should have all the tools needed.

- Part I:
  - How many countries are present?
  - What's the timespan of the data?
  - Does each country have a row for every year present?
  - Which countries are missing years?
  - How many rows have at least 1 non-null values across the columns 3-57?

- Part II - Convert this to a long format with the following columns:
  1. country
  2. year
  3. diagnosis method
  4. gender: male or female.
  5. age: lower_age - higher_age, e.g., 0-14
  6. number of cases
- Part III:
  - Create a graph that shows when countries started to report TB cases.