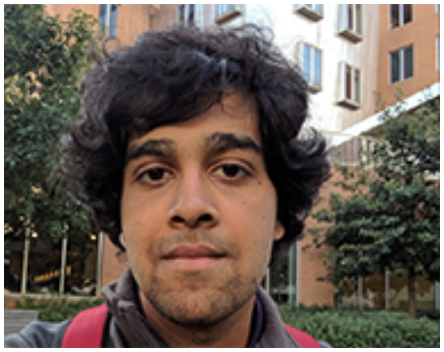
Computational Systems Biology
Deep Learning in the Life Sciences

6.802 20.390 20.490 HST.506
6.874 Area II TQE (AI)

David Gifford
Lecture 1
February 4, 2019

http://mit6874.github.io

Your guides



Sid Jain
sj1@mit.edu

Konstantin Krismer
krismer@mit.edu

Saber Liu
geliu@mit.edu

http://mit6874.github.io

mit6874.github.io

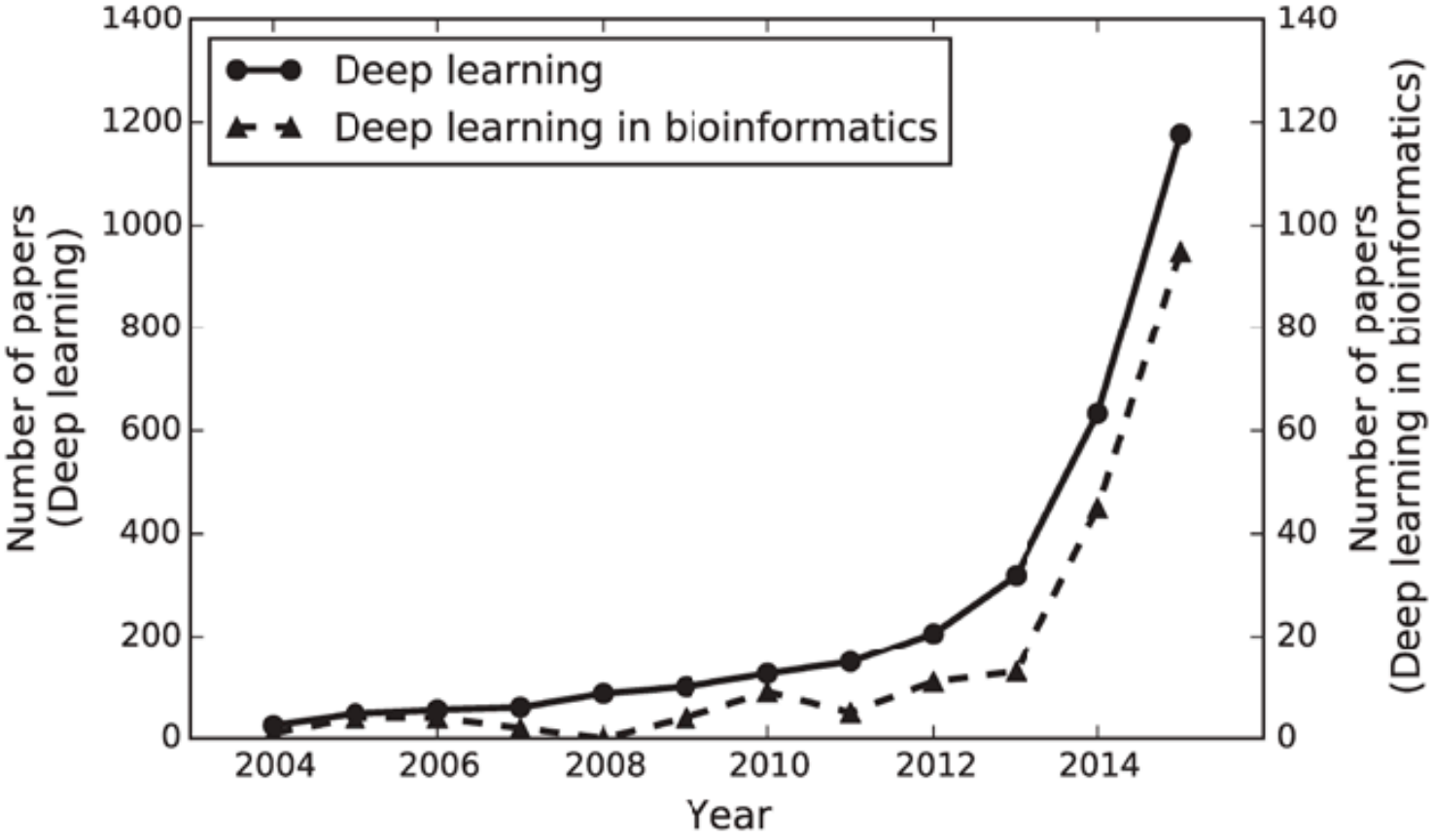
6.874staff@mit.edu

You should have received the Google Cloud coupon URL in your email

Recitations (this week)
Thursday 4 - 5pm 36-155
Friday 4 - 5pm 36-155

Office hours are after recitation at 5pm
in same room
(PS1 help and advice)

Approximately 8% of deep learning publications are in bioinformatics

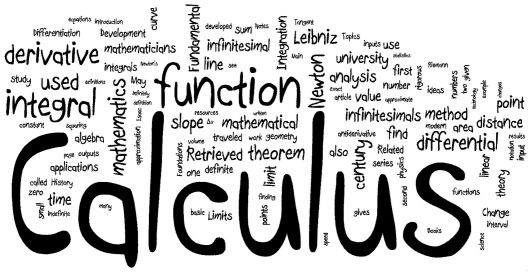


Welcome to a new approach to life sciences research

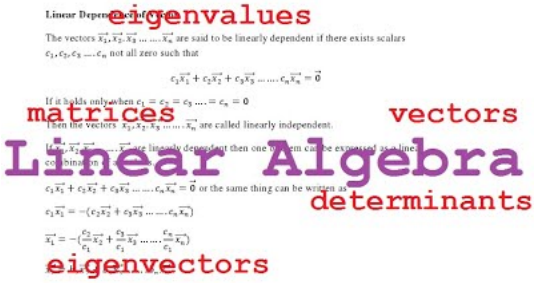
- Enabled by the convergence of three things
  - Inexpensive, high-quality, collection of large data sets (sequencing, imaging, etc.)
  - New machine learning methods (including ensemble methods)
  - High-performance Graphics Processing Unit (GPU) machine learning implementations
- Result is completely transformative

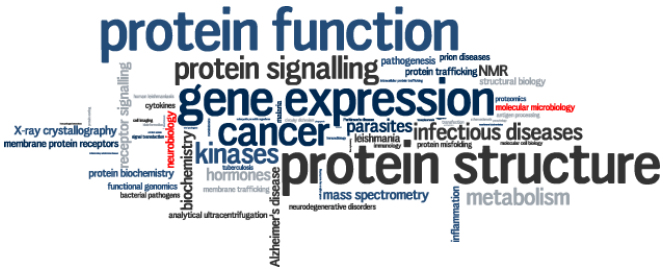
Your background

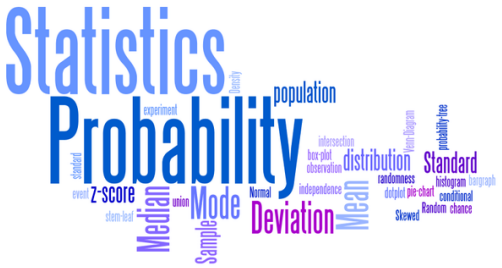
- Calculus, Linear Algebra
- Probability, Programming
- Introductory Biology




```
def loadstable(ver):
    return _loadversion(ver, prefix="_stable_")

def loadunstable(ver):{...}
def loadexact(ver):{...}
def _loadversion(ver, prefix):
    targetname = prefix + ver.replace('.', '_')
    mainpackage = _original__import__("tools", globals(), locals(),
        [targetname])
    global currentversion
    currentversion = getattr(mainpackage, targetname)

    # Let users change versions after choosing this one
    currentversion.loadstable = loadstable
    currentversion.loadunstable = loadunstable
    currentversion.loadexact = loadexact

    return currentversion

currentversion = None
```







Alternative introductory MIT subjects

- 6.047 / 6.878 Computational Biology: Genomes, Networks, Evolution
- 8.592 Statistical Physics in Biology
- 7.09 Quantitative and Computational Biology
- 7.32 Systems Biology
- 7.33 Evolutionary Biology: Concepts, Models and Computation
- 7.57 Quantitative Biology for Graduate Students
- 18.417 Introduction to Computational Molecular Biology
- 20.482 Foundations of Algorithms and Computational Techniques in Systems Biology


This subject is the red pill

https://arxiv.org/abs/1710.10196

Machine Learning is the ability to improve on a task with more training data

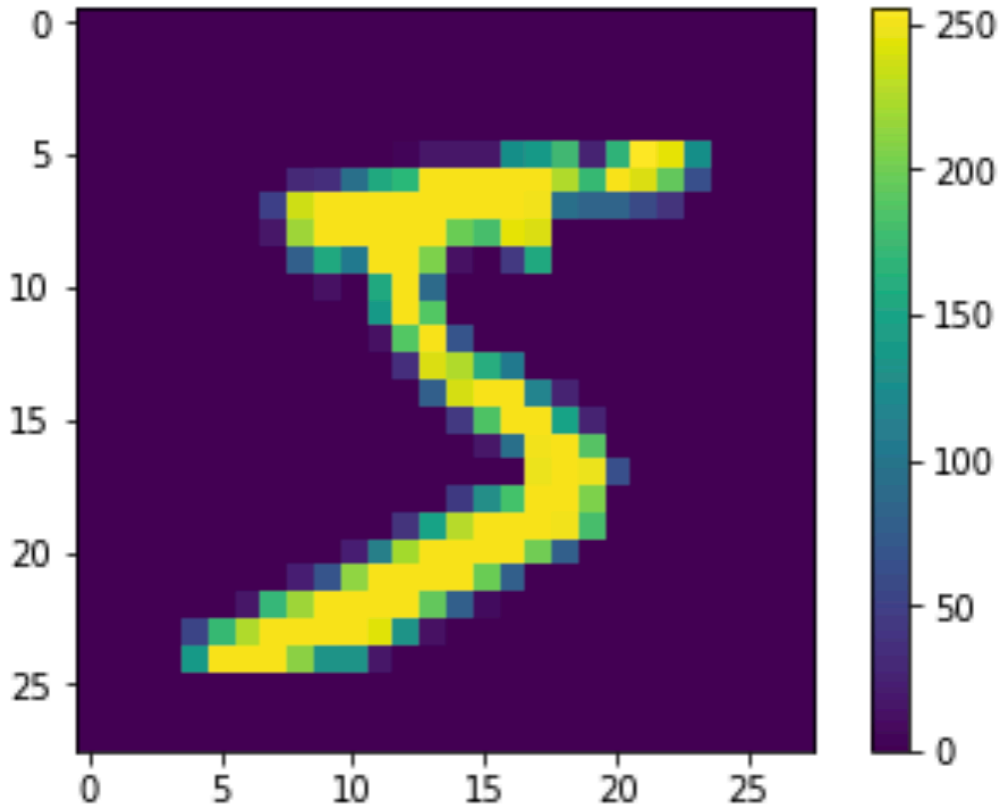
- Task T to be performed
  - Classification, Regression, Transcription, Translation, Structured Output, Anomaly Detection, Synthesis, Imputation, Denoising
- Measured by Performance Measure P
- Trained on Experience E (Training Data)

Welcome

L 1	Feb. 5	Machine learning in the computational life sciences
L 2	Feb. 7	Neural networks and TensorFlow
R 1	Feb 7	Machine Learning Overview and PS 1
L 3	Feb 12	Convolutional and recurrent neural networks

Problem Set: Softmax MNIST (PS 1)

PS 1: Tensor Flow Warm Up



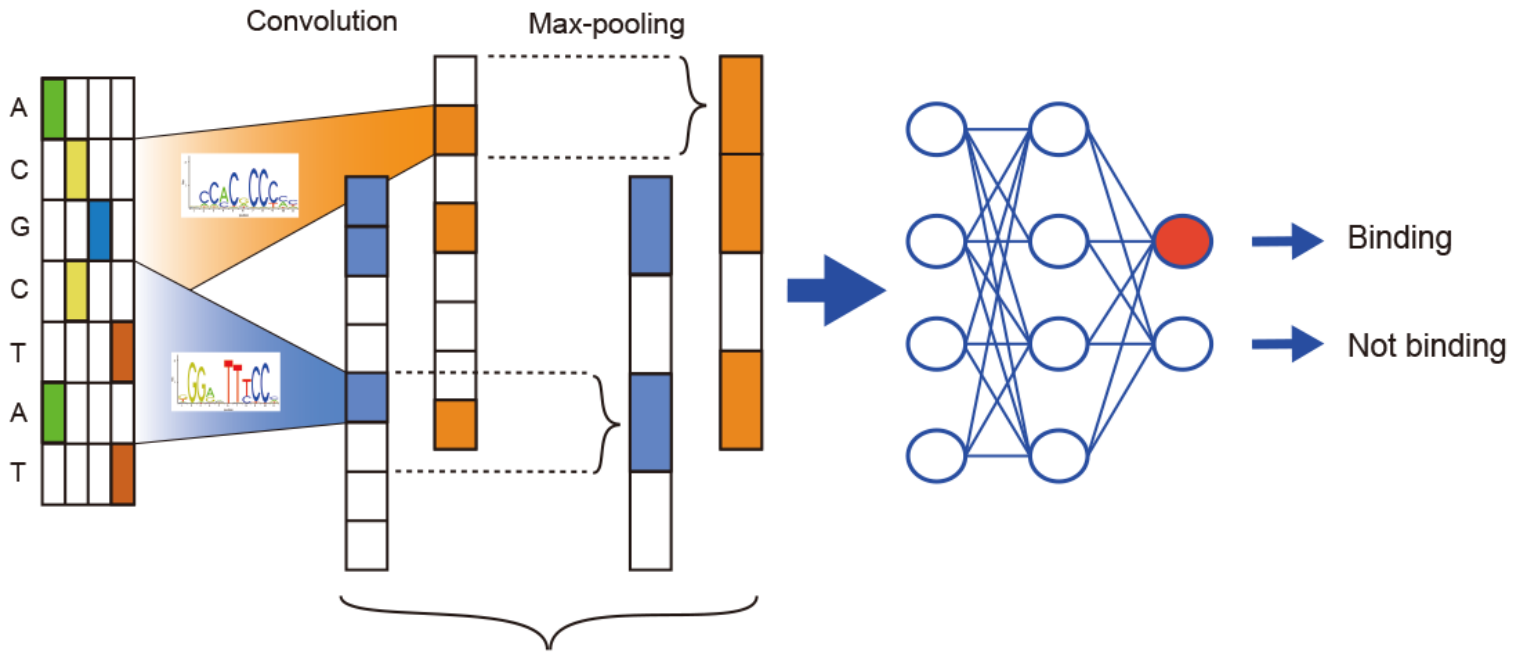

```
ground truth: 5
```

Regulatory Elements / ML models and interpretation

L 4	Feb 14	Protein-DNA interactions
R 2	Feb. 14	Neural Networks and TensorFlow
	Feb. 19	(Holiday - President's Day)
L 5	Feb. 21	Models of Protein-DNA Interaction
R 3	Feb. 21	Motifs and models
L 6	Feb. 26	Model interpretation (Gradient methods, black box)

Problem Set: Regulatory Grammar

PS 2: Genomic regulatory codes



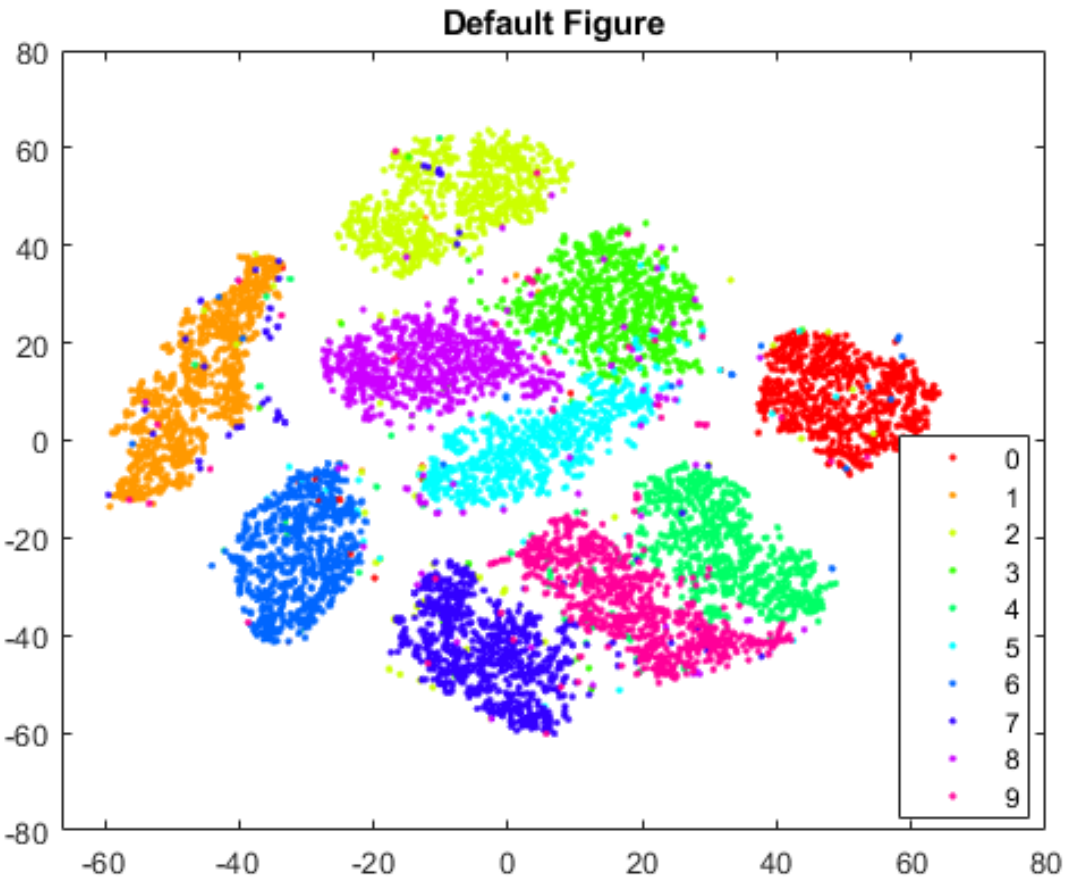
One-hot encoded sequence context → Stacks of convolutional and max-pooling layers → Fully-connected layers → TF binding prediction

The Expressed Genome / Dimensionality reduction

L 7	Feb. 28	The expressed genome and RNA splicing
R 4	Feb 28	Model interpretation
L 8	Mar 5	PCA, dimensionality reduction (t-SNE), autoencoders
L 9	Mar 7	scRNA seq and cell labeling
R 5	Mar 7	Compressed state representations

Problem Set: scRNA-seq tSNE

PS 3: Parametric tSNE

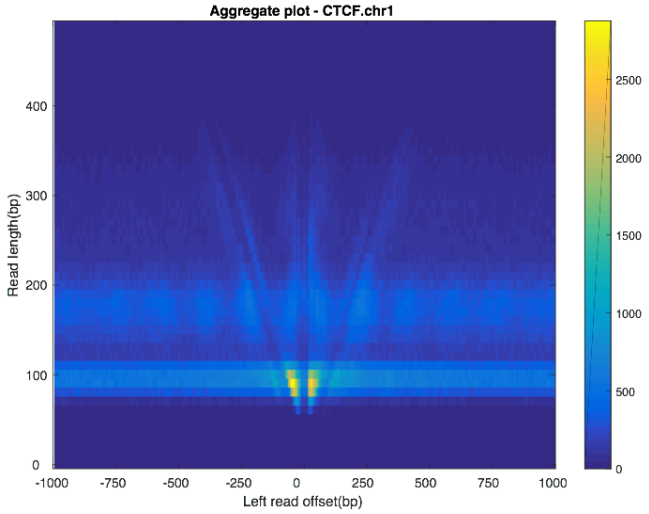


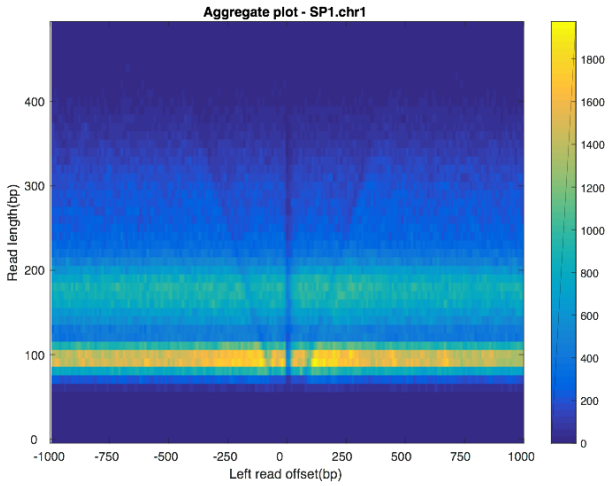
Gene Regulation / Model selection and uncertainty

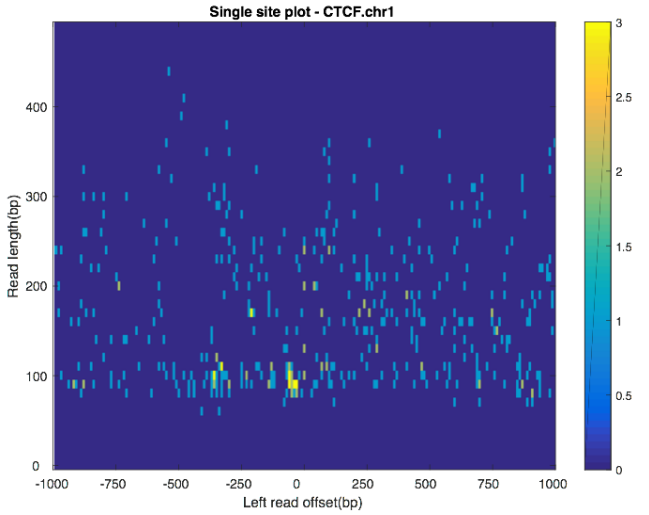
L 10	Mar 12	Modeling gene expression and regulation
L 11	Mar 14	Model uncertainty, significance, hypothesis testing
R 6	Mar 14	Model selection and L1/L2 regularization
L 12	Mar 19	Chromatin accessibility and marks
L 13	Mar 21	Predicting chromatin accessibility
R 7	Mar 21	Chromatin accessibility

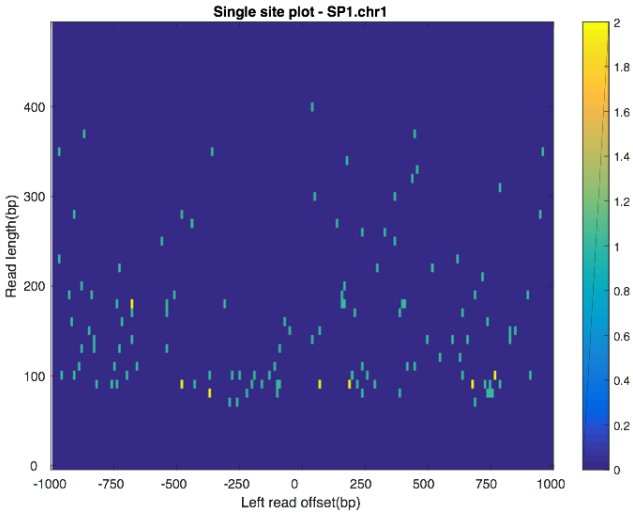
Problem Set: CTCF Binding from DNase-seq

PS 4: Chromatin Accessibility









Genotype -> Phenotype, Therapeutics

L 14	Apr 2	Discovering and predicting genome interactions
L 15	Apr 4	eQTL prediction and variant prioritization
R 8	Apr 4	Lead SNPs to causal SNPs; haplotype structure
L 16	Apr 9	Imaging and genotype to phenotype
L 17	Apr 11	Generative models: optimization, VAEs, GANs
R 9	Apr 11	Generative models
L 18	Apr 18	Deep Learning for eQTLs
L 19	Apr 23	Therapeutic Design
L 20	Apr 25	Exam Review
L 21	Apr 30	Exam

Problem Set: Generative models for medical records

PS 5: Generative Models

Sample 1: discharge instructions: please contact your primary care physician or return to the emergency room if [*omitted*] develop any constipation. [*omitted*] should be had stop transferred to [*omitted*] with dr. [*omitted*] or started on a limit your medications. * [*omitted*] see fult dr. [*omitted*] office and stop in a 1 mg tablet to tro fever great to your pain in postions, storale. [*omitted*] will be taking a cardiac catheterization and take any anti-inflammatory medicines diagness or any other concerning symptoms.

Your programming environment

Problem 2

In this problem, we wish to use CNN to learn the motif of CTCF from sequences with similar di-nucleotide frequency. The positive samples are 101bp sequences centered at CTCF ChIP-seq peaks from GM12878 cell line. The negative sequences are generated by permuting the nucleotides in the positive sequences while keeping the di-nucleotide frequency.

We will provide functions for loading data, training and testing. You will:

- implement a CNN model with given specifications
- specify the initilization of parameters in the model
- train the model and evaluate on the test set

All the places where you need to fill in begins with "TODO" and ends with "END OF YOUR CODE".

In [1]:
```
import tensorflow as tf, sys, numpy as np, h5py
from os.path import join,dirname,basename,exists,realpath
from os import makedirs
from tensorflow.examples.tutorials.mnist import input_data
from sklearn.metrics import roc_auc_score
```

In [2]:
```
data_folder = '../data/motif_disc'
batch_size = 128
valid_size = 2000
epochs = 20
best_model_file = join('../output',basename(data_folder),'best_model.ckpt')
if not exists(dirname(best_model_file)):
    makedirs(dirname(best_model_file))
```

In [3]:
```
# Function to load the data embedded in the previous problem and their labels
def load_data(mydir):
    train = h5py.File(join(mydir,'train.h5'),'r')
```


Your computing resource

Cloud Platform Education Grants

Use credits provided to you via the Google Cloud Platform Education Grants program to access Google Cloud Platform. Get what you need to build and run your apps, websites and services.

Thank you for your interest in Google Cloud Platform Education Grants. Please fill out the form below to receive a coupon code for credit to use on Google Cloud Platform.

First Name

Last Name

School Email

@mit.edu

If you do not see your domain listed, please contact your course instructor: gifford@mit.edu

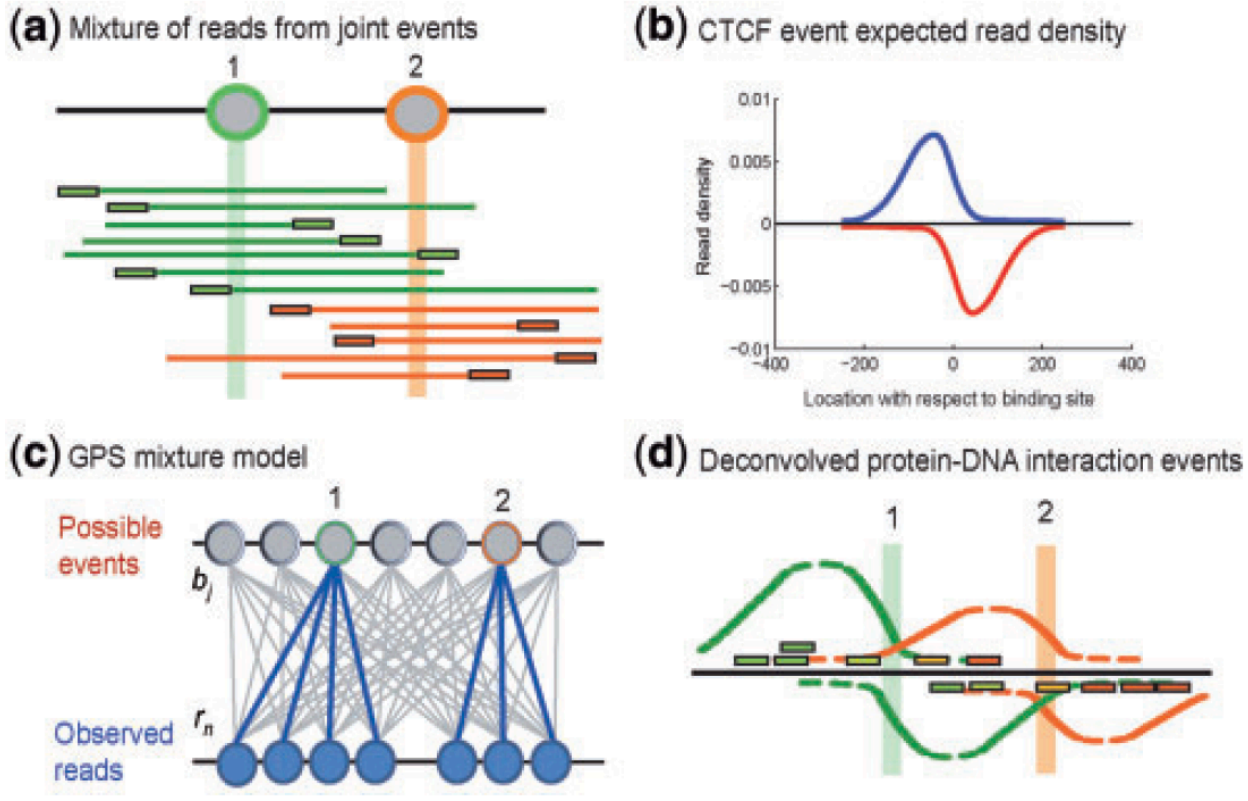
By clicking "Submit" below, you agree that we may share the following information with your educational institution and course instructor (gifford@mit.edu): (1) personal information that you provide to us on this form and (2) information regarding your use of the coupon and Google Cloud Platform products.

Submit

Your grade is based on 5 problem sets, an exam, and a final project

- Five Problem Sets (40%)
  - Individual contribution
  - Done using Google Cloud, Jupyter Notebook
- In class exam (1.5 hours), one sheet of notes (30%)
- Final Project (30%)
  - Done individually or in teams (6.874 by permission)
  - Substantial question

ML resolves Protein-DNA binding events



- Who - what protein(s) are binding?
- Where - where are they binding?
- Why - what chromatin state and sequence motif causes their binding?
- When - what differential binding is observed in different cell states or genotypes?
- How - are accessory factors or modifications of the factor involved?

How can we establish ground truth?

- Replicate experiments should have consistent observations
- Independent tests for same hypothesis (different antibody, different assay)
- Statistical test against a null hypothesis - what is the probably of seeing the reads at random? We need a *null model* for this test.

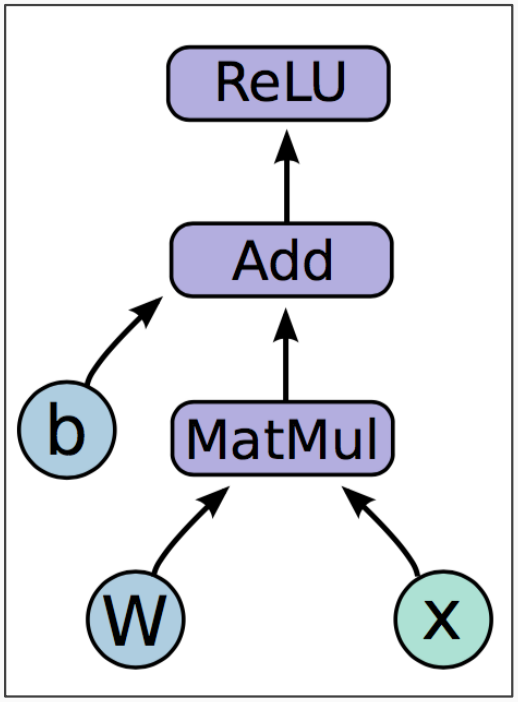
Programming model

Big idea: Express a numeric computation as a **graph**.

- Graph nodes are **operations** which have any number of inputs and outputs
- Graph edges are **tensors** which flow between nodes

Programming model: NN feedforward

$$h_i = \mathrm{ReLU}(Wx + b)$$

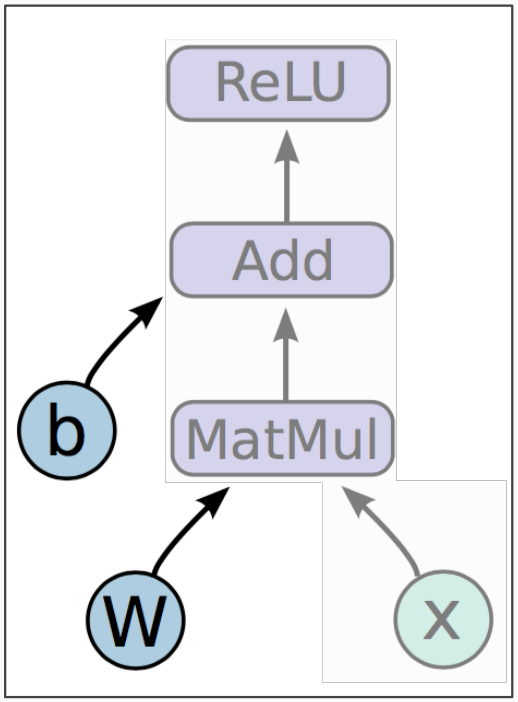


Programming model: NN feedforward

$$h_i = \mathrm{ReLU}(Wx + b)$$

Variables are 0-ary stateful nodes which output their current value.
(State is retained across multiple executions of a graph.)

(parameters, gradient stores, eligibility traces, …)

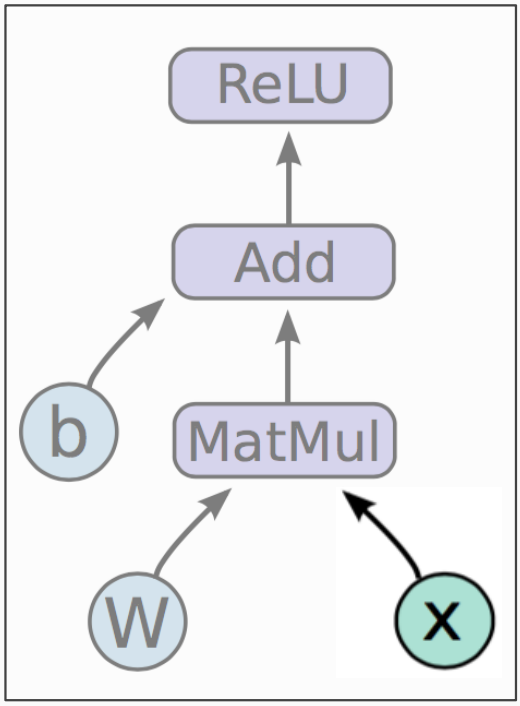


Programming model: NN feedforward

$$h_i = \mathrm{ReLU}(Wx + b)$$

Placeholders are 0-ary nodes whose value is fed in at execution time.

(inputs, variable learning rates, …)



Programming model: NN feedforward

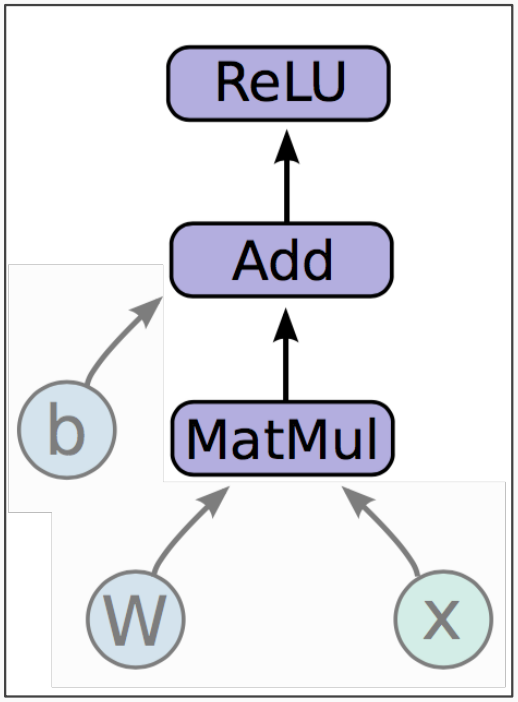
$$h_i = \mathrm{ReLU}(Wx + b)$$

Mathematical operations:

MatMul: Multiply two matrix values.

Add: Add elementwise (with broadcasting).

ReLU: Activate with elementwise rectified linear function.

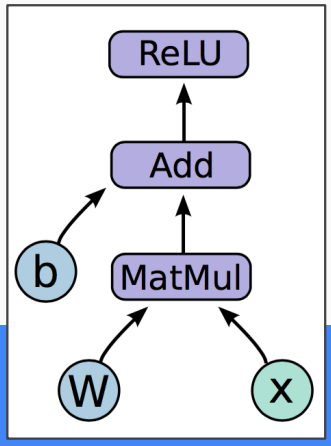


In code, please!

1. Create model weights, including initialization
    a. `W` ~ *Uniform*(-1, 1); `b` = **0**
2. Create input placeholder `x`
    a. *m* * 784 input matrix
3. Create computation graph

```
import tensorflow as tf

1 b = tf.Variable(tf.zeros((100,)))
  W = tf.Variable(tf.random_uniform((784,
  100), -1, 1))
2 x = tf.placeholder(tf.float32, (None,
  784))
3 h_i = tf.nn.relu(tf.matmul(x, W) + b)
```

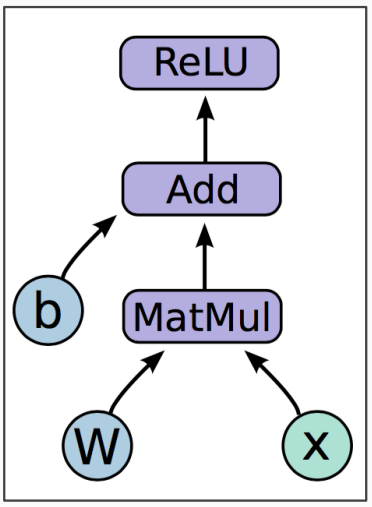
$$h_i = \mathrm{ReLU}(Wx + b)$$



How do we run it?

So far we have defined a **graph**.

We can deploy this graph with a **session**: a binding to a particular execution context (e.g. CPU, GPU)



Getting output

`sess.run(fetches, feeds)`

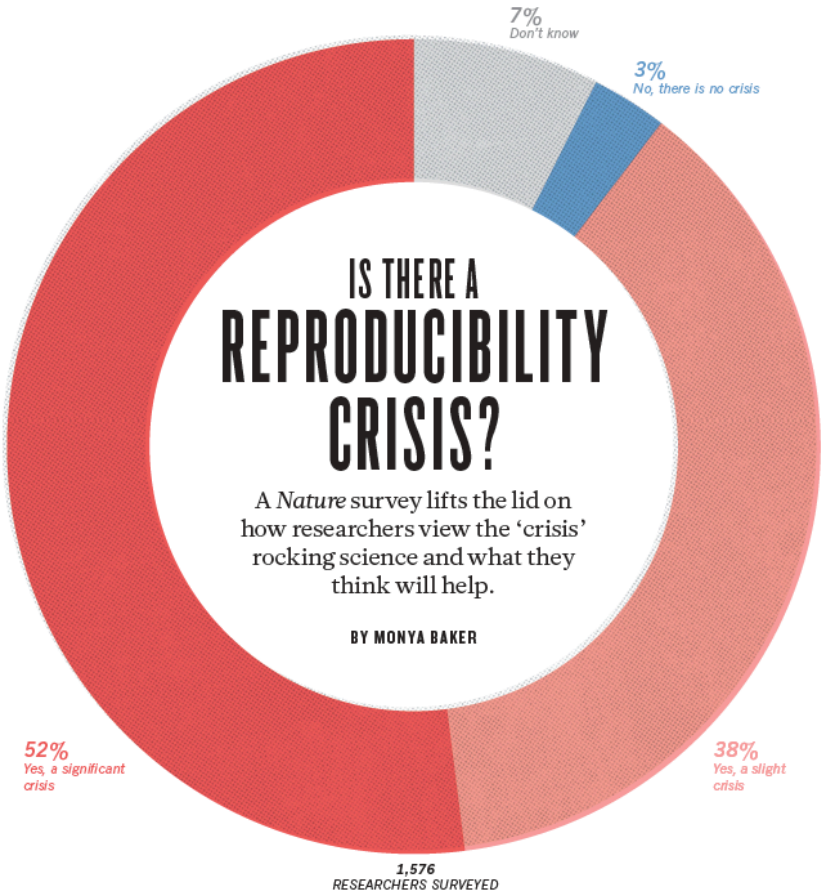
Fetches: List of graph nodes. Return the outputs of these nodes.

Feeds: Dictionary mapping from graph nodes to concrete values. Specifies the value of each graph node given in the dictionary.

```
import numpy as np
import tensorflow as tf

1 b = tf.Variable(tf.zeros((100,)))
  W = tf.Variable(tf.random_uniform((784,
  100) -1, 1))
2 x = tf.placeholder(tf.float32, (None, 784))
3 h_i = tf.nn.relu(tf.matmul(x, W) + b)


  sess = tf.Session()
  sess.run(tf.global_variables_initalizer
  ())
  sess.run(h_i, {x: np.random.random(64,
  784)})
```

Basic flow

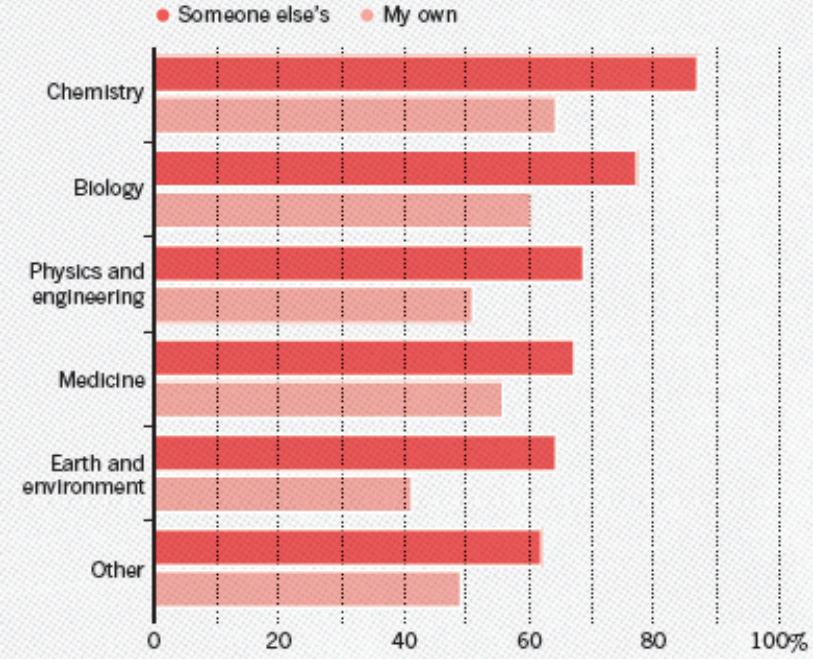
1. Build a graph
    a. Graph contains parameter specifications, model architecture, optimization process, …
    b. Somewhere between 5 and 5000 lines
2. Initialize a session
3. Fetch and feed data with `Session.run`
    a. Compilation, optimization, etc. happens at this step — you probably won't notice



http://www.nature.com/polopoly_fs/1.19970!/menu/main/topColumns/topLeftColumn/pdf/533452a.pdf

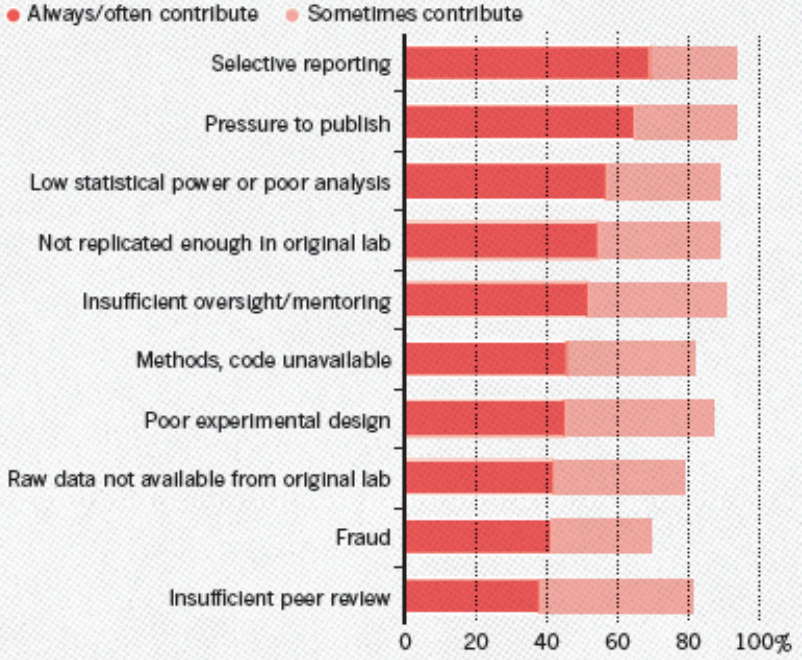
HAVE YOU FAILED TO REPRODUCE AN EXPERIMENT?

Most scientists have experienced failure to reproduce results.



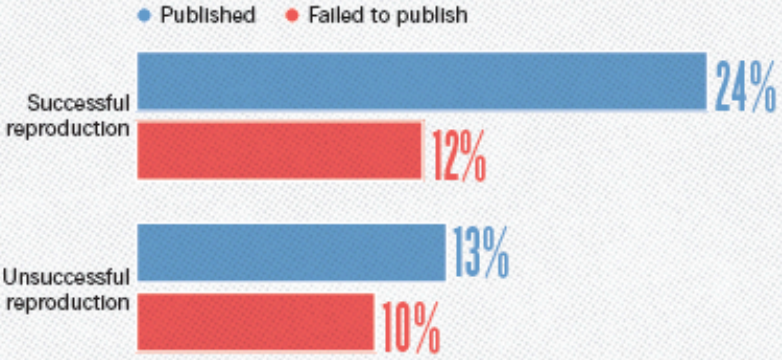
WHAT FACTORS CONTRIBUTE TO IRREPRODUCIBLE RESEARCH?

Many top-rated factors relate to intense competition and time pressure.



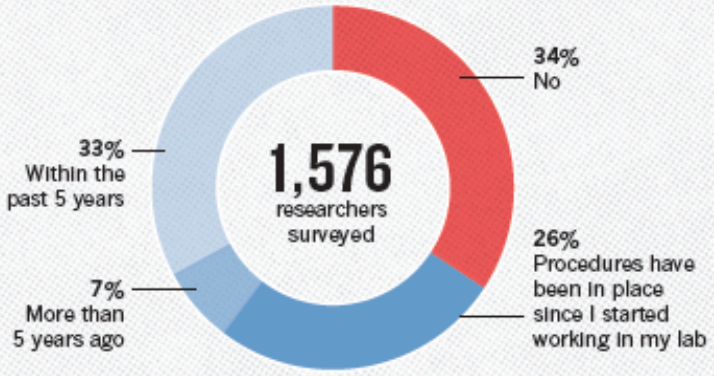
HAVE YOU EVER TRIED TO PUBLISH A REPRODUCTION ATTEMPT?

Although only a small proportion of respondents tried to publish replication attempts, many had their papers accepted.



HAVE YOU ESTABLISHED PROCEDURES FOR REPRODUCIBILITY?

Among the most popular strategies was having different lab members redo experiments.



Number of respondents from each discipline:
Biology **703**, Chemistry **106**, Earth and environmental **95**, Medicine **203**, Physics and engineering **236**, Other **233**

Amgen could not reproduce the findings of 47/53 (89%) landmark preclinical cancer papers

REPRODUCIBILITY OF RESEARCH FINDINGS

Preclinical research generates many secondary publications, even when results cannot be reproduced.

Journal impact factor	Number of articles	Mean number of citations of non-reproduced articles*	Mean number of citations of reproduced articles
>20	21	248 (range 3–800)	231 (range 82–519)
5–19	32	169 (range 6–1,909)	13 (range 3–24)

Results from ten-year retrospective analysis of experiments performed prospectively. The term 'non-reproduced' was assigned on the basis of findings not being sufficiently robust to drive a drug-development programme.
*Source of citations: Google Scholar, May 2011.

http://www.nature.com/nature/journal/v483/n7391/pdf/483531a.pdf

Direct and conceptual replication is important

- **Direct replication** is defined as attempting to reproduce a previously observed result with a procedure that provides no a priori reason to expect a different outcome
- **Conceptual replication** uses a different methodology (such as a different experimental technique or a different model of a disease) to test the same hypothesis; tries to avoid confounders

https://elifesciences.org/content/6/e23383

Reproducibility Project: Cancer Biology
Registered Report/Replication Study Structure

- A **Registered Report** details the experimental designs and protocols that will be used for the replications, and experiments cannot begin until this report has been peer reviewed and accepted for publication.
- The results of the experiments are then published as a **Replication Study**, irrespective of outcome but subject to peer review to check that the experimental designs and protocols were followed.

https://elifesciences.org/content/6/e23383

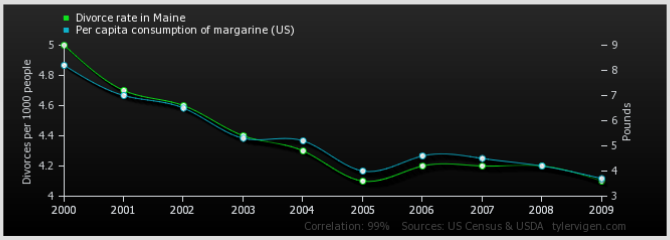
Claim precision is key to science

- “We have discovered the regulatory elements”
- “We have predicted the regulatory elements”

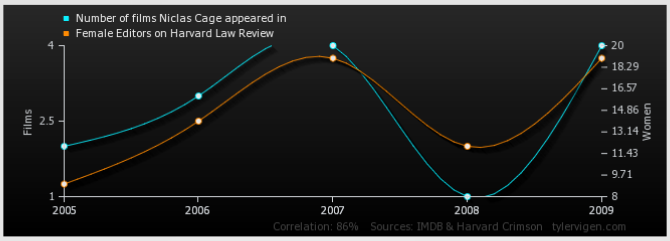
- “The variant causes a difference in gene expression”
- “The variant is associated with a difference in gene expression”

Correlation ≠ Causation

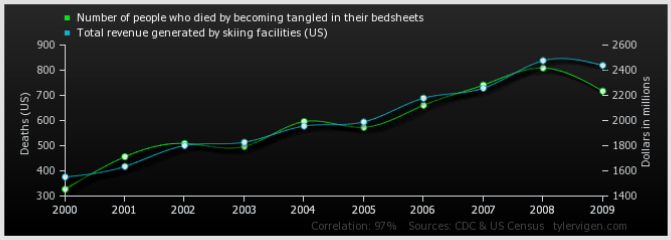
Divorce rate in Maine
correlates with
Per capita consumption of margarine (US)



Number of films Niclas Cage appeared in
correlates with
Female Editors on Harvard Law Review

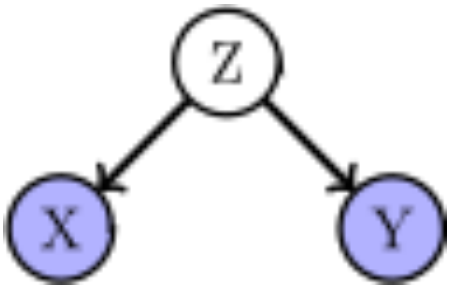


Number of people who died by becoming tangled in their bedsheets
correlates with
Total revenue generated by skiing facilities (US)



Interventions enable causal statements

- Observation only data can be influenced by confounders
- A confounder is an unobserved variable that explains an observed effect
- Interventions on a variable allow for the detection of its direct and indirect effects




This subject is the red pill