# Kung Faux Pandas
## Alternative Facts for Privacy Protection

**James King & Seth Russell**
Data Science to Patient Value (D2V)
School of Medicine
University of Colorado Anschutz Medical Campus
Aurora, CO
james.king@ucdenver.edu
seth.russell@ucdenver.edu

## Abstract

A description of an end-to-end system for easily generating faux-data which is statistically similar to given real data but does not contain sensitive personal information. This system makes enables data to be distributed for reproducibility testing and other purposes while in full compliance with HIPAA and GDPR.

## 1 Introduction

*"Because it works, bitches."*

– Richard Dawkins on his preference for the scientific method over other modes of inquiry.

The ability of independent investigators to verify results is perhaps *the* most fundamental feature that separates science from other forms intellectual inquiry. Unfortunately, the mechanisms of communicating scientific results have changed little since the age of enlightenment and are inadequate enabling reproducibility for some modern scientific endeavors such as machine learning. Perhaps the most important impediment to reproducibility is the need to share data. This has long been difficult in health care fields, but will presently become difficult for just about everyone.

There is a large body of published literature detailing methods for overcoming this difficulty via "anonymizing" and "synthesising" data [**?** **?** ], however at present a mechanism to routinely *use* these techniques has not been forthcoming.

This paper details an open source software library called *Kung Faux Pandas (KFP)* which integrates *any* of these data security methods into the popular Python Pandas data science library. For those who prefer other tools, KFP also provides Structured Query Language (SQL) interface which enables users to arbitrarily query any data within a database, returning to the user a data set which either does not contain any personal information (synthesized) or has been anonymized by standard methods.

## 2 Background

In the United States, there's been a long-standing regulation described in the Health Insurance Portability and Accountability Act (HIPAA), which mandates strong security on data related to health care and is backed by significant fines and possible criminal charges [**?** ]. Thus, even institutitions which which to share data face a steep institutional risk. This makes it virtually impossible to enable independent verification of machine learning (ML) models which often require enormous amounts of training data.

31st Conference on Neural Information Processing Systems (NIPS 2017), Long Beach, CA, USA.

Perhaps more ominously for researchers, on May 25 , 2018, the European Union's (EU) General Data Protection Regulation (GDPR) has come into full effect [**?** ], establishing a new paradigm in data "ownership" for that jurisdiction. In short, this set of rules defines the "owners" of data to be people from whom it was collected and obliges anyone possessing that data to honor the owners' preferences about how the data is kept and used, including a requirement to deletion of any data at the request of its owner at any time in the future.

These rules are ultimately good for citizens, but they create a pickle for scientists studying humans as it's very difficult to follow normal processes in scientific rigor while complying with privacy protection rules.

One obvious way around this pickle is to separate data and analysis in such a way that the analysis can be performed on data without the *analyst* having access to the data. This is fairly straight-forward to implement. Essentially, the analyst, with only a description of the sensitive data, composes analytical code which is handed off to the data custodian. The custodian then runs it and returns the results to the researcher.

This has been technologically possible for some time, but it's awkward in practice because activities such as data cleaning, exploration, and plotting are interactive and iterative processes which would become impractical with a mediator.

Kung Faux Pandas sidesteps this problem by generating *faux-data* that is "statistically similar" to the real data but does not itself contain any sensitive data.

This faux-data can be prodded, poked, plotted, and posted for the world to see, enabling analysts to develop their code using whatever means they prefer. When the analytical code is ready, it can then be sent to the data custodian to run on the real data.

## 3 Methods

Many methods of generating the faux-data exist in the literature, many of which have software implementations freely available. KFP provides a standard mechanism for "wrapping" any method into a "plug-in" which integrates the method with the Pandas data frame model. Three of these plug-ins are provided with full documentation for creating other plug-ins.

### 3.1 Included Methods

| Method | Source | Plug-In Name | Notes |
|---|---|---|---|
| Kernel Density Estimator | scipy.stats.gaussian_kde | KDEPlugin | Useful for inter-related ordinal and ratio data |
| DataSynthesizer | <authors> | DataSynthesizerPlugin | Generates a Baysean tree to capture some column relationships |
| CMF? | <authors> | Can't Remember | Fast |

Describe KFP's structure, including the "plugin" mechanism.

Describe the SQL interface

Diagrams - not sure if we need both of these. Should we include a screenshot of the UI (saved to /images)?

Figure 1: System Architecture.

Figure 2: Data synthesis process.

## 4 Conclusion

Basically need to put abstract here...