Quinn, Barry

Working Paper

Teaching Open Science Analytics in the Age of Financial Technology

QMS Research Paper, No. 2022/01

Provided in Cooperation with:
Queen's University Belfast, Queen's Business School

Suggested Citation: Quinn, Barry (2022) : Teaching Open Science Analytics in the Age of Financial Technology, QMS Research Paper, No. 2022/01, Queen's University Belfast, Queen's Management School, Belfast,
https://doi.org/10.2139/ssrn.4019430

This Version is available at:
https://hdl.handle.net/10419/271256

QUEEN'S UNIVERSITY BELFAST | MANAGEMENT SCHOOL

Working Paper Series - QMS Research Paper 2022/01

Teaching open science analytics in the age of financial technology

Barry Quinn

Queen's University Belfast

27 January 2022

Series edited by Philip T. Fliers and Louise Moss.
To check for updated versions of this paper here.
To subscribe click here.
To submit forward your paper to qms.rps@qub.ac.uk.

Teaching open science analytics in the age of financial technology

A Preprint

Barry Quinn *
Queens Management School
Queen's University Belfast
Belfast
b.quinn@qub.ac.uk

January 28, 2022

Abstract

We consider the challenge of teaching open science analytics in finance in the computer age. There is a crisis of confidence in science; especially finance. We argue that the unstoppable algorithmic tranformation of financial services, and the nascent field of financial machine learning provide an opportunity to redesign finance programmes for the age of financial technology. We argue it is time for a rethink how we can extract reliable statistical inference from financial data given proliferation of computing, *Big financial data*, and the unstoppable algorithmisation the finance industry. The paper begins by agnostically profiling the modelling paradigm choice. Next we establish the developments in statistical inference in the computer age specific to finance. Finally, we consider the idea of placing computation as a central tennent in finance curicullum, and discuss the infrastructure and tools involved. We illustrate a use case where the infrastrcture is on-boarded in a cloud computing suite with enterprise-level server software. We are not arguing that finance is computation, rather that by placing computation as a frictionless part of the curicullum, students can engage with the full suite of state-of-the-art inferential tools avaliable to financial data science practicioners.

***K*eywords** Open source analytics · Finance education · Financial technology · Statistical inference · Financial data science · Financial data science and machine learning · Econometrics · Cloud computing · Employability

1 Introduction

The open science movement is a global initiative to combat a crisis of confidence in scientific research. This credibility crisis spans many fields, including medicine (Ioannidis, 2005), psychology (Nosek et al., 2012), management (Bettis, 2012), experimental economics (Maniadis et al., 2017), and financial economics (Kelly et al. 2022). In business schools, this crisis has two distinct but interconnected parts. Firstly, a crisis of confidence in knowledge published in academic journals and, by extension, a crisis of relevance in the classroom. The mission is to provide education guided by research(Responsible Research in Business & Management 2020). This paper seeks to understand the challenge of teaching responsible analytics in the fast movement age of financial technology.

The algorithmic transformation of financial services has seen the financial technology (FinTech) industry surge from the sidelines to the mainstream[2]. This surge is especially true in the United Kingdom, where FinTech is

*A special thanks to Dr Alan Hanna for his insightful comments.

[2] *A progress report on fintech's record-breaking year* by Nicholas Megaw August 2021. https://www.ft.com/content/89ea3d5d-cd29-46ec-88f1-67729b09a7c2?shareType=nongift

viewed as **a permanent technology revolution that is changing the way we do finance**[3]. FinTech is multidisciplinary and is moving from the technological era of *social, mobile, analytics and cloud*(SMAC) to a future where *distributed ledger technology, artificial intelligence, extended reality, Quantum computing*(DARQ) technologies well displace SMAC. For students to remain relevant in this fast-paced world, computation to enable reliable inference must play a central role in curricula.

While finance is a social science, many parts of modern finance are fundamentally quantitative, with financial practitioners solving practical problems using innovative technologies. Furthermore, the rise of big and alternative data combined with the exponential growth of AI and financial data science has created new opportunities in the financial sector. AI and machine learning applications are now widespread and include innovations in risk management (Lin and Hsu 2017), portfolio construction (Jaeger et al. 2021), investment banking (Investment Banking Council 2020) and insurance (Society of Actuaries 2020). In short, the *algorithmisation* of finance is unstoppable (López de Prado 2019).

Statistical inference is a broad discipline at the intersection of mathematics, empirical science and philosophy. Since its philosophical beginnings through the publication of the Bayes rule in 1763[4], computation has been a traditional bottleneck for applied statistical inference frameworks, motivating small sample solutions with solid asymptotic principles (Efron and Hastie 2016). Traditional econometrics retained much of this framework arguable because of the sparsity of observed data realisation of theory. Until the early 1950s, the computation bottle still dominated small sample solutions in applied statistics. Nevertheless, as power and accessibility of computing have increased, and statistical theory has developed, statistical inference using machine learning models has become commonplace for applied statisticians[5]

While narrow AI, which uses rule-based algorithms, has dominated the fast-paced automation of tasks in financial services, researchers predict the next wave of automation will be the digitising judgement calls (López de Prado 2018). Given that finance professionals have an essential fiduciary duty towards their clients, the rapid growth of artificial intelligence (AI) in finance has highlighted some critical risks around trust, overfitting, lack of interpretability, biased inputs and unethical use of data. Now more than ever, highly computationally digitally literate finance graduates are needed to balance algorithmic technology developments with sustainability, ethics, bias, and privacy to create *trustworthy* data-driven decisions (Mahdavi and Kazemi 2020).

The UK is leading the way in FinTech innovation and is forging on with a large scale plan post-Brexit. The 2021 Kalifa Review on FinTech sets out an ambitious five-point plan to foster and scale UK based FinTech firms. A central part of this plan is to *upskill*, and *reskill* adults by developing training and courses from high-quality universities. So now more than ever, there are exciting opportunities for computationally literate finance graduates in the UK.

This paper provides an overview of the opportunities and challenges for the financial education curricula in the fast-paced world of technology innovation in business. We specifically focus on; (1) open science statistical inference, a complementary blend of traditional econometric inference and the emerging field of financial machine learning; (2) how to embed computation to facilitate a frictionless approach to teaching open science statistical inference. Finally, we provide an overview of how this has been achieved in the Management School of Queens University Belfast using an enterprise-scale cloud computing infrastructure and a suite of enterprise-level web software.

2 Background

2.1 What is Statistical Inference?

We use statistical inference to learn from incomplete or imperfect data. Formally, statistical inference is a set of operations on data that yield estimates and uncertainty statements about predictions and parameters of some underlying process or population. Mathematically, these probabilistic uncertainty statements are derived based on some assumed probability model for observed data. Understanding errors in statistical work depends on inferential statements built on the concepts of probability modelling, estimation, bias and variance. A central theme in modern statistical inference is uncertainty. However, mainstream statistics has

[3]Kalifa Review of UK FinTech 2021, `https://www.gov.uk/government/publications/the-kalifa-review-of-uk-fintech`

[4]Which was used by early advocates to argue the existence of God.

[5]One notable example is the *bootstrap* a computer-intensive inferential engine that is now ubiquitous in applied statistics.

long argued that it is a mistake to use hypothesis tests or statistical significance to attribute certainty from noisy data Gelman, Hill, and Vehtari (2020).

There are three standard paradigms for thinking about statistical inference:

- The *sampling model* is used to learn characteristics about the population (mean and standard deviation of all possible realisations of Apple INC stock returns), which we must estimate from a sample, or subset, of that population.
- The *measurement error model* helps us learn aspects of some underlying pattern or law (usually described in regression format where the coefficients are our targets of learning, for example; $y_i = a + bx_i$), but the data is measured with error (the prevalent form in finance is $y_i = a + bx_i + \epsilon$, less common are models with measurement error in x)
- *Model error* refers to the inevitable imperfections of the models that we apply to real-world financial data.

What students find challenging in a traditional econometrics class because these paradigms are different, and in practice, we often consider all three when building statistical models for finance. For instance, consider the regression model predicting asset price returns using the common pricing factors constructed from data. There is typically a sampling aspect to such a study, performed on some subset of asset returns to generalise to a larger population. The model includes measurement error, at least implicitly, as the common factors usually have some form of measurement error in their construction (Hang et al. 2019) and indeed model error because any assumed functional form can only be approximate. Financial markets are complex networks that vary by time and circumstance; this variation can be a measurement or model error.

2.2 What is financial machine learning (hereafter FML)?

Machine learning (hereafter ML) proliferates many real-world applications. Still, it has been slow to develop in areas of scientific research, especially economic analysis, where traditional econometric techniques dominate. Athey and Imbens (2019) argue this is due to a clashing culture, where some financial economists view the ontological differences between econometrics and machine learning as intractable. This naive comparison highlights the epistemological challenges computer age statistical inference faces in a world of rapid algorithmic development (Efron and Hastie 2016). FML is a subfield of AI in its infancy, attempting to reconcile the differences between econometrics and ML.

ML is a branch of nonparametric statistics mixing statistical learning, computer science and optimisation (Molina and Garip 2019), where algorithms have three fundamental building blocks:

1. A loss function;
2. An optimisation criteria;
3. An optimisation routine.

Changes in each of these building blocks produce a wide variety of learning algorithms characterising their freedom to learn patterns in the data. ML algorithms are categorised into unsupervised learning and supervised learning. A classic example of the former is clustering, and the latter is a regression tree. A learning algorithm with no feedback is unsupervised in that the analyst provides no information to guide the learning process. In contrast, supervised learning involves feedback in the form of training data that is correctly labelled. Other types of machine learning are prevalent in finance between these two extremes. For example, reinforcement learning uses partial feedback, in the form of *rewards*, to encourage the desired behaviour without instructing the algorithm precisely (Dixon and Polson 2020).

In the classical sense, ML models are statistically biased. Due to their optimisation of a restricted objective according to a specific algorithmic methodology and statistical rationale. On the other hand, econometrics applies statistics to a data sample, usually in the form of regression analysis, to examine relationships. The model design uses well-journeyed economic theory to develop an *unobservable* hypothesised model. The asymptotic theory is then relied upon to produce objective statistical inference, which minimises bias, possibly at the expense of increased sampling variation.

FML attempts to reconcile three broad conflicts between ML and econometrics(Lommers, Harzli, and Kim 2021):

1. The importance of statistical inference and modelling paradigm;

2. Causality;
3. An a priori hypotheses and model assumptions.

In what follows, we consider each of these conflicts in turn. We begin with the choice of estimation models.

2.3 Modelling paradigm

To move from computation to inference in statistics, we must make an estimation choice. Mathematical statistics is the science of learning from experience that arrives a little at a time (Efron and Hastie 2016) and using this information to quantify uncertainty and variation(Spiegelhalter 2019). Unlike many other disciplines in mathematics, there is no unifying theory. Reliable inference from statistics requires careful thought beyond the computing algorithm. The modern financial data scientist is faced with two abstracting *leaps of faith* to go from computation to meaningful inference. The theoretical inference paradigms are classical (or frequentist) and bayesian .[6]. This section aims to present a disinterested perspective on both paradigms and guide when and why each choice is preferred.

2.3.1 Frequentist inference

With its developmental beginnings in 1900, frequentism has grown to dominate 20^{th} century statistical inference in finance. A remarkably potent theory, it was primarily designed to produce maximally efficient statistical analysis using small data collected by hand under strictly controlled conditions.

The name *frequentism* seems to have been suggested by Neyman as a statistical analogue of Richard von Mises' frequentist theory of probability; the connection is made explicit in his 1977 paper, "Frequentist probability and frequentist statistics." "Behaviourism" might have been a more descriptive name since the theory revolves around the long-run behaviour of statistics $t(x)$. That said, *frequentism* has stuck, replacing the older (sometimes disparaging) term *objectivism* (Efron and Hastie 2016).[7]

Statistical inference usually begins with the assumption that some probability model has produced the observed data x, $x = (x_1, x_2, .., x_n)$. Let $X = (X_1, X_2, ..., X_n)$ indicate n independent draws from a probability distribution F, written:

$$F \to X$$

F is the underlying distribution of possible prices (the model). The statistician observes a realisation $X = x$ of $F \to X$, and then wishes to infer some property of the unknown distribution F . Suppose the desired property is the expectation of a single random draw X from F, denoted:

$$\theta = E_f \{X\}$$

Otherwise, there is room for error and *the inferential question is how much error?*. The estimate $\hat{\theta}$ using some algorithm. Importantly, $\hat{\theta}$ is a realisation of $\mathbf{\Theta} = \mathbf{t}(\mathbf{X})$ the output of t(.) applied to some theoretical sample X from F. It follows that frequentist inference focuses on the accuracy of an observed estimate $\hat{\theta} = t(x)$, which represents >the probabilistic accuracy of $\mathbf{\Theta} = \mathbf{t}(\mathbf{X})$ as an estimator of θ. T

His proposition contains the powerful idea that $\hat{\theta}$ is just a number, but $\hat{\Theta}$ takes a range of values whose spread can define measures of accuracy.

Bias and variance Bias and variance are familiar examples of frequentist inference. Defining μ to be the expectation of $\hat{\Theta} = t(X)$ under the model $F \to X$:

$$\mu = E_F \left\{ \hat{\Theta} \right\}$$

[6]Roughly speaking, frequentists infer meaning by asking themselves *what would I see if I reran the same situation (and again and again and again,. . . , ad infinitum, et ultra)?*. On the other hand, bayesian coax a fantastical belief that *they have prior knowledge of the situation, encode this in probability, and update this knowledge by learning from a set of observed data points.* In essence, one needs to be somewhat of a fantasist to generalise from statistical measurement.

[7]Neyman's attempt at a complete frequentist theory of statistical inference, *inductive behaviour*, is not much-quoted today but can claim to influence Wald's development of decision theory.

The bias attributed to estimate $\hat{\theta}$ of parameter θ is

$$bias = \mu - \theta$$

. The variance attributed to estimate $\hat{\theta}$ of parameter θ is:

$$var = E_f\left\{(\hat{\Theta} - \mu)^2\right\}$$

Importantly, what keeps this from being a tautology, and is one of its biggest Bayesian criticisms, is that attribution to the *single number* $\hat{\theta}$ of the probabilistic properties of $\hat{\Theta}$ derived from the model

$$F \to X$$

.

Formally, frequentism is defined as *an infinite sequence of future trials.* We imagine hypothetical datasets $X^{(1)}; X^{(2)}; X^{(3)}$... generated by the same mechanism as x providing corresponding values $\hat{\Theta}^{(1)}; \hat{\Theta}^{(2)}; \hat{\Theta}^{(3)}$.... The frequentist principle is then to attribute for $\hat{\theta}$ the accuracy properties of the ensemble of $\hat{\Theta}$ values. As mentioned above, in essence, frequentists ask themselves, *What would I see if I reran the same situation (and again and again)....?*

In practice, there is an apparent defect in this principle. It requires the calculation of the properties of the estimators $\mathbf{\Theta} = \mathbf{t}(\mathbf{X})$ obtained from the actual distribution F, even though F is unknown. In practice, frequentism uses a collection of ingenious devices to circumvent this defect, including the plugin principle[8], Taylor series approximations[9], parametric families and maximum likelihood theory, simulation and the bootstrap[10], and pivotal statistics[11]

The popularity of frequentist methods reflects their relatively modest mathematical modelling assumptions: only a probability model F (more exactly a family of probabilities) and an algorithm of choice. Such flexibility has some defects. Primarily, the principle of frequentist correctness does not help with the choice of algorithm. That is frequentist need to find the *best*(optimal) choice of $t(x)$ given model F. In the early 1900s, two theories emerged. 1. Fisher's theory of maximum likelihood: in specific parametric probability models, the MLE is the optimum estimate in terms of the minimum(asymptotic) standard error. 2. Neyman-Pearson lemma provides an optimum hypothesis-testing algorithm.

2.4 Bayesian inference

A human mind is an inference machine: *It is getting windy, the sky is darkening, I had better bring my umbrella.* Unfortunately, it is not a dependable machine, especially when weighing complicated choices against experience. Nevertheless, Bayes' theorem is a surprisingly simple mathematical guide to accurate inference. The theorem (or *rule*), now 250 years old, marked the beginning of statistical inference as a serious scientific subject. The theorem has varied in influence over the centuries, now in the ascendance due to computer-age algorithms and inference.

If not directly opposed to frequentism, Bayesian inference is at least orthogonal. It reveals some worrisome flaws in the frequentist point of view while at the same time exposing itself to the criticism of dangerous overuse. The struggle to combine the virtues of the two philosophies has become more acute in an era of massively complicated data sets. Here we will review some basic Bayesian ideas and how they impinge on frequentism.

Roughly, the Bayesian statistical model can be thought of as a model for learning from data but regulated by domain knowledge. While labelled as both frequentist or bayesian in flavour, machine learning models can be framed comfortably within this definition.

[8]The frequentist accuracy estimate for the mean of x plugs in as an estimate of the variance of a single X draw from F into a formula relating the standard error to this said variance.

[9]Statistics more complicated than a simple average can often be related to the plugin formula by local linear approximation sometimes know as the *delta method*.

[10]modern computation has opened up the possibility of numerically implementing the *infinite sequence of future trails* except for the endless part.

[11]These are statistics whose distribution does not depend upon the underlying probability distribution F. A classic example is Student's two-sample t-test statistic

In finance, Bayesian statistics are not as popular as the classical modelling paradigm. More recently, though, in a bid to resolve the puzzle of the replication (credible) crisis in financial economics research, leading figures a being to embrace Bayesian inference as a more credible alternative in the face of data dredging (p-hacking) criticisms (See Jensen, Kelly, and Pedersen 2021 for a detailed exposition on a how Bayesian inference improve both internal and external validity in finance research)

2.4.1 Bayesian inference as counting possibilities

Modestly, Bayesian inference is just counting and comparison of possibilities. Bayesian inference uses a concept similar to Jorge Luis Borges short story The Garden of Forking Paths. Borges explores all paths in this book, with each decision branching outward into an expanding garden of forking paths. This is the same device that Bayesian inference offers.

A sensible approach to inference about what happened is to consider everything that could have happened. A Bayesian analysis is a garden of forking data in which alternative sequences of events are cultivated. Some of these alternative sequences are pruned as we learn about what happened. In the end, what remains is only what is logically consistent with our knowledge. This approach provides a quantitative ranking of hypotheses, a ranking that is maximally conservative, given the assumptions and data that go into it.

This approach cannot guarantee a correct answer on real-world terms. However, it can guarantee the best possible answer, on fantasy world terms, that could be derived from the information fed into it.

Suppose there is a bag, and it contains four marbles. These marbles come in two colours: blue and white. We know there are four marbles in the bag, but we do not know how many each colour is. We do know that there are five possibilities:

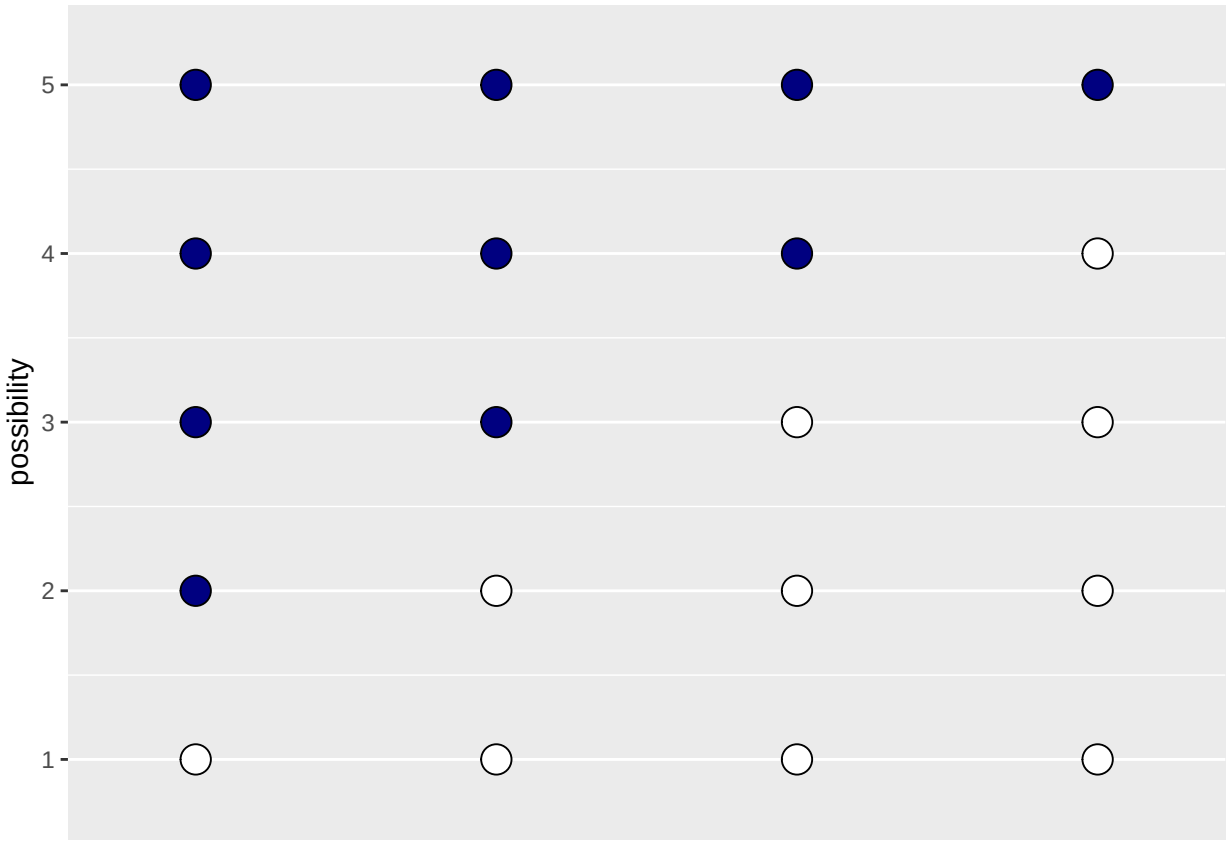

Figure 1: All the possible draws from a bag containing 2 white and 2 blue

These are the only possibilities consistent with what we know about the bag's contents. Call these five possibilities the conjectures (a collection of hypotheses). Our goal is to figure out which of these conjectures is most plausible, given some evidence about the bag's contents.

Enter the evidence: a sequence of three marbles is pulled from the bag, one at a time, replacing the marble each time and shaking the bag before drawing another marble. The sequence that emerges is: blue, white, blue in that order. These are the data. So now, let us use the data to infer what is in the bag. Let us begin by considering the single conjecture that the bag contains one blue and three white marbles. After three draws, there are 64 possible paths (4^3), but as we consider each draw from the bag, some paths are logically eliminated.

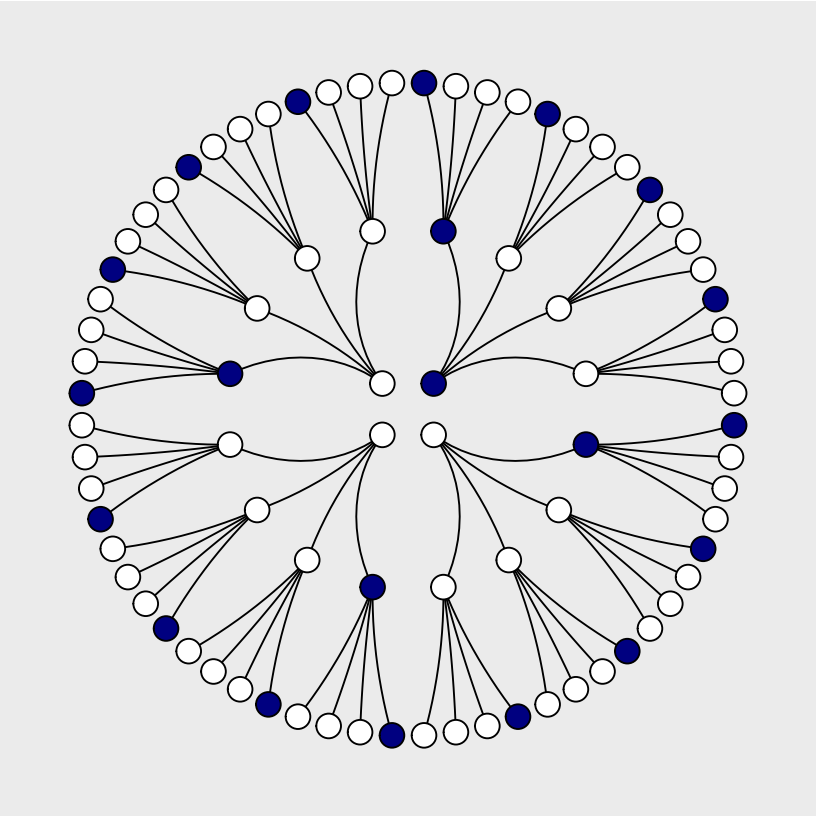

Figure 2: All possible pathways

]

We can then eliminate the paths inconsistent with the observed sequence. The first draw turned out to be blue; imagine the actual data tracing out a path through the garden; it must have passed through the one blue path near the origin. The second draw from the bag produces a white marble, so three of the paths forking out of the first blue marble remain—finally, the third draw in blue. Visually, we can see that logically eliminating the other paths leaves three ways for the sequence to appear, assuming the bag contains [blue, white, white, white].

p_1	p_2	p_3	p_4	draw 1: blue	draw 2: white	draw 3: blue	ways to produce
w	w	w	w	0	4	0	0
b	w	w	w	1	3	1	3
b	b	w	w	2	2	2	8
b	b	b	w	3	1	3	9
b	b	b	b	4	0	4	0

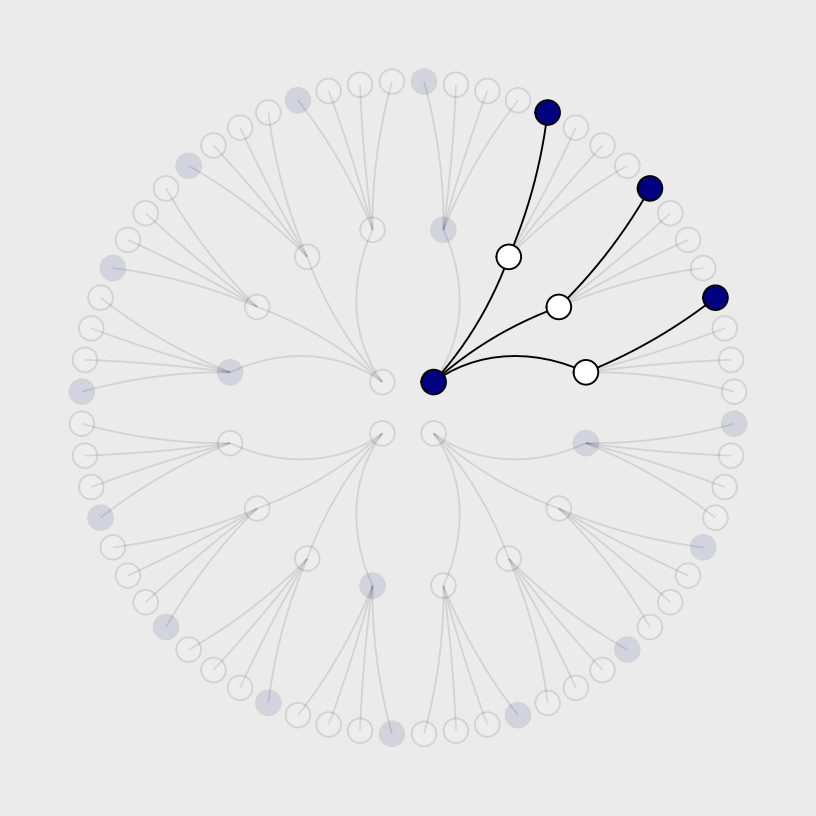

Figure 3: Eliminating the inconsistent pathway given the data

To summarise, we have considered five different conjectures about the bag's contents, ranging from zero blue marbles to four blue marbles. For each of these conjectures, we have counted how many sequences paths through the garden of forking data could potentially produce the observed data [blue, white,blue].

It is noteworthy that the number of ways to produce the data for each conjecture can be computed by counting the number of paths in each *ring* of the garden and then multiplying these counts together. Note that multiplication is just counting condensed. This point will come up again when we look at the formal representation of Bayesian inference. We can use these counts to rate the relative plausibility of each conjecture. Luckily, there is a mathematical way to compress all of this. Specifically, we define the updated plausibility of each possible composition of the bag, after seeing the data, as:

Plausibility of [bwww] after seeing bwb $\propto$ ways [bwww] can produce [bwb] $\times$ prior plausibility of [bwww]

Probability can be thought of as plausibility standardise and if we helpfully define p=1/4 (the proportion of blue marbles)

A conjectured proportion of blue marbles, p, is usually called a *parameter* value. It is just a way of indexing explanations of the data. In social science, conjectures usually come as linear regression parameters added together as learning targets for an observed outcome variable. The relative number of ways a value p can produce the data is usually called a *likelihood*. It is derived by enumerating all the possible data sequences

p_1	p_2	p_3	p_4	p	ways to produce data	plausibility
w	w	w	w	0.00	0	0.00
b	w	w	w	0.25	3	0.15
b	b	w	w	0.50	8	0.40
b	b	b	w	0.75	9	0.45
b	b	b	b	1.00	0	0.00

that could have happened and then eliminating those inconsistent with the data. The prior plausibility of any specific p is usually called the *prior probability*. The new, updated plausibility of any specific p is usually called the *posterior* probability.

2.4.2 Units of statistical inference

For both Bayesians and frequentists the fundamental unit of statistical inference are probability densities:

$$F = \{f_u(x); x \in X, \mu \in \Omega\};$$

Where x, the observed data, is a point in the sample space X, while unobserved parameter μ is a point in the parameter space Ω. A statistician observes x from $f_u(x)$, and infer the value of μ ˆ[Popular families of distributions include

1. the Normal family:

$$f_u(x) = \frac{1}{\sqrt{2\pi}} e^{-0.5(x-\mu)^2}$$

helpful when we want X and Ω being on the entire natural line $(-\infty, \infty)$

2. Poisson family:

$$f_u(x) = e^{-\mu)} \mu^{-} x/x!$$

useful when X is a nonnegative integer $\{0, 1, 2, ...\}$ and Ω is the nonnegative real number line $(0, \infty)$]. In addition, Bayesian inference requires one crucial assumption, the knowledge of a prior density concerning the parameter $g(\mu), \mu \in \Omega$. $g(\mu)$ represents preliminary information concerning the parameter μ, available to the statistician *before* the observation of x. Exactly what constitutes **prior knowledge** is a crucial and contentious question in financial econometrics.

Roughly speaking, Bayesian inference is about counting probabilities. The rule is a simple exercise in conditional probability. Formally, $g(\mu|x) = g(\mu) f_\mu(x)/f(x)$, where f(x) is a marginal density (an integral or a sum of discrete where we count up all the possibilities). In this rule, x is fixed at its observed value while μ varies over Ω. **This is the opposite of frequentist calculations**. ˆ[A memorable restatement of this rule is that the posterior odds ratio is the prior odds ratio, time the likelihood ratio. Formally, this is defined for any two points μ_1 and μ_2 on Ω as:

$$\frac{g(\mu_1|x)}{g(\mu_2|x)} = \frac{g(\mu_1)}{g(\mu_1)} \frac{f_{\mu_1}(x)}{f_{\mu_2}(x)}$$

(Efron and Hastie 2016)]

2.4.3 Comparing modelling paradigms

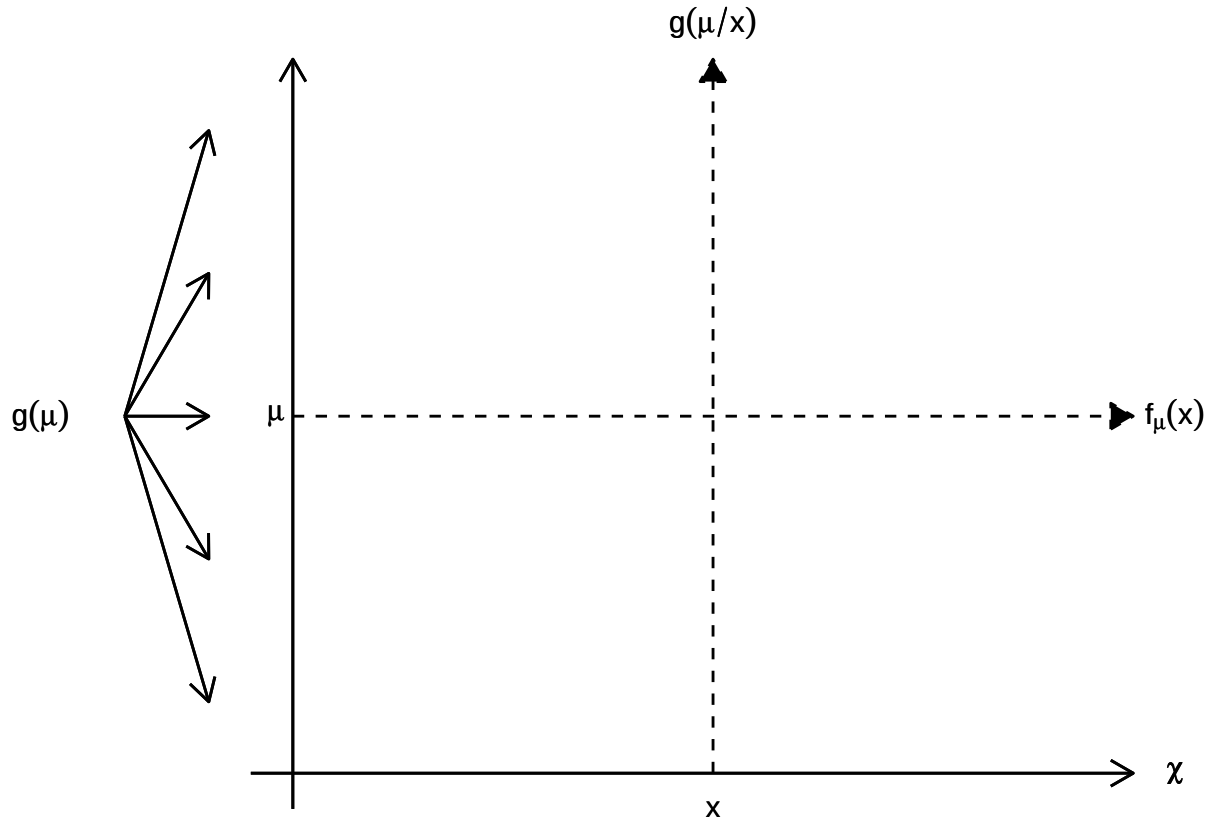

Figure 4: Bayesian inference proceeds vertically given x; frequentist inference proceeds horizontally, given mu

Bayesians and frequentists start on the same playing field, a family of probability distributions $f_\mu(x)$, but play the game in orthogonal directions. Figure @(fig:compare) indicated schematically in Bayesian inference proceeds vertically, with x fixed, according to the posterior distribution $g(u|x)$. Frequentists reason horizontally, with fixed μ and x varying. There are advantages and disadvantages accrue to both strategies.

Bayesian inference requires a prior distribution $g(\mu)$. When experience provides $g(\mu)$, there is every good reason to employ Bayes' theorem. If not, techniques such as those of Jeffreys still permit the use of Bayes' rule, but the results lack the full logical force of the theorem. The Bayesian's right to ignore selection bias, for instance, must then be treated with caution. Frequentism replaces the choice of a prior with the choice of a method or algorithm, $t(x)$, designed to answer the specific question at hand. This arbitrary choice in the inferential process can lead to contradictions. The optimal choice of $t(x)$ reduces arbitrary behaviour. However, computer-age applications typically move outside the safe waters of classical optimality theory, lending an ad-hoc character to frequentist analyses.

Modern data-analysis workflows typically favour a methodology, such as logistic regression or regression tree. This approach plays into the methodological orientation of frequentism, which is more flexible than Bayes' rule in dealing with specific algorithms. One always hopes for a reasonable Bayesian justification for the method at hand.

Having chosen $g(\mu)$ only a single probability distribution $g(\mu|x)$ is in play for Bayesians. Frequentists, by contrast, must struggle to balance the behaviour of $t(x)$ over a family of possible distributions since μ in Figure 3.5 is unknown. The growing popularity of Bayesian applications (usually begun with uninformative priors) reflects their simplicity of application and interpretation. The simplicity argument cuts both ways. The Bayesian essentially bets it all on the choice of their prior being correct, or at least not harmful. Frequentism takes a more defensive posture, hoping to do well, or at least not poorly, whatever μ might be.

A Bayesian analysis answers all possible questions at once. Frequentism focuses on the problem at hand, requiring different estimators for different questions. This is more work but allows for a more intense inspection of particular problems. The simplicity of the Bayesian approach is especially appealing in dynamic contents, where data arrives sequentially and updating one's beliefs is a natural practice. Financial market dynamics are a case in point. Bayes' theorem is an excellent tool for combining statistical evidence from disparate sources, the closest frequentist analogue being maximum likelihood estimation.

In the absence of genuine prior information, a whiff of subjectivity hangs over Bayesian results, even those based on uninformative priors. Classical frequentism claimed for itself the high ground of scientific objectivity, especially in contentious areas such as drug testing and approval, where sceptics and friends hang on the statistical details. Figure 3.5 is soothingly misleading in its schematics: In FML, μ and x have typically been high-dimensional, sometimes very high-dimensional, straining to the breaking point of both the frequentist and the Bayesian paradigms.

Computer-age statistical inference at its most successful combines elements of the two philosophies, as in the empirical Bayes methods or the lasso.

There are two potent arrows in the statistician's philosophical quiver, and faced, say, with 1000 parameters and 1,000,000 data points, there is no need to go hunting armed with just one of them.

2.5 Rebooting econometrics

Statistical inference is the bedrock of econometrics, while the main focus of ML is prediction. In traditional econometrics, models learn statistical pictures about the *unobservable* data generating process parameters. Their power emanates from an *a priori* probability model under strict assumptions with a proven track record. Armed with this theoretical confidence and using the dominant frequentist approach, econometricians can objectively infer uncertainty and variation characteristics about **how well the data sampled maps to the theoretical data generating process**.

Econometricians coax validate statistical inference using amenable distributional assumptions and model specifications. The three most important properties in most traditional econometrics models are linearity, additivity and monotonicity. However, the most crucial assumption, routine overlooking in many textbooks, is **validity**. Andrew Gelman summarises this property as:

> The data you are analysing should map to the research question you are trying to answer. This assumption sounds obvious but is often overlooked or ignored because it can be inconvenient. Optimally, this means that the outcome measure should accurately reflect the phenomenon of interest. The model should include all relevant predictors. The model should generalise to the cases to which it will be applied. - (Gelman, Hill, and Vehtari 2020)

These amenable formulations provide a convenient root to statistical significance using p-values (Lommers, Harzli, and Kim 2021), but the inherent philosophy of traditional econometric models is incompatible with out-of-sample inference and prediction (López de Prado 2019); two tasks which are at the core of the modern finance industry.

In contrast, machine learning models focus on outcome prediction, where the data generated process is generally undefined, to algorithmically optimise models to fit the underlying data generating process as well as possible (Lommers, Harzli, and Kim 2021). (Efron and Hastie 2016) summaries this well in their definition of computer age statistical inference:

> Very broadly speaking, algorithms are what statisticians do, while inference says why they do them. However, the efflorescence of ambitious algorithms has forced an evolution (though not a revolution) in inference, the theories by which statisticians choose among competing methods.

Thus the challenge for today's finance graduates is to understand the inferential benefits of machine learning in the rigorous setting of econometrics. For inference to be convincing, more work must be done on the statistical consistency of machine learning models. For instance, in Explainable AI (XAI), great strides have been made to produce statistical consistent and cognitive convincing explanations of the importance of predictors in the ML model (Barredo Arrieta et al. 2020). For instance, in recent years, there have been notable advances. For example, second-generation p-values can be included in a penalised regression model to yield tangible advantages for balancing support recovery, parameter estimation, and prediction tasks (Blume et al. 2019; Zuo, Stewart, and Blume 2021).

2.5.1 Causality

Identifying causal effects with data has a long and varied history. It's origins span many disciplines, including early statisticians (Fisher 1936), economists (Haavelmo 1943; Rubin 1974), geneticists (Wright 1934), and even computer scientists (Pearl 2009). We can view causal inference as using theory and expert institutional knowledge to estimate the impact of events or decisions on a given outcome of interest (Cunningham 2021). A naive assumption would be that prediction algorithms in ML cannot provide the rigour of empirical econometric design in extracting causal inference. However, a growing sub-field of ML tackles causality in two ways. Firstly, it can improve the predictive power of traditional econometrics by decoupling the search for relevant predictors from the search for specification (López de Prado 2018). Secondly, machine learning can play a key role in discovering new financial theories beyond traditional methods, such as a new theory in market microstructure that explained the 2010 flash crash (Easley et al. 2020).

2.5.2 Hypotheses, assumptions and cultural clashes

Traditionally, machine learning is data-driven, while econometrics is hypothesis-driven. Valid inference from testing stands on model assumptions being asymptotically ground truth. Over 20 years ago, the Berkeley statistician, Leo Breiman, lambasted the statistical community for their dogmatic approaches in the face of emerging algorithmic techniques to statistical science successes. He framed his argument as a culture problem where

> ..the statistical community has been committed almost exclusively to data models... where one assumes that a given stochastic data model generates the data. (Breiman 2001)

For the most part, the statistical community has now accepted machine learning (ML) as a standard part of statistical science, with graduate-level standards incorporating ML techniques alongside the traditional statistical approaches (Hastie, Tibshirani, and Friedman 2009; Efron and Hastie 2016) and leading statisticians exposing their benefits for enhancing scientific discovery (Spiegelhalter 2019).

While the statistics community has moved on, the economics and econometrics community has been much slower to depart from the strictness of data-generating models which embody consistency, normality and efficiency. The econometric canon pre-dates the dawn of digital computing, with models devised for estimation by hand. These are legacy technologies that need updating for the digitally savvy graduates of the future.

ML approaches do not naturally deliver these theoretical properties^ [Technically, the No Free lunch theorem applies has been applied to machine learning (Wolpert and Macready 1997). This states that `a a  priori` no one learning algorithm can be defined as the *best* performer. Machine learning experts have argued that relevance of this criticism in recent years as research in statistical inference in machine learning develops [Giraud-Carrier, Christophe, and Foster Provost. "Toward a justification of meta-learning: Is the no free lunch theorem a show-stopper." In Proceedings of the ICML-2005 Workshop on Meta-learning, pp. 12–19. 2005.; Whitley, Darrell, and Jean-Paul Watson. "Complexity theory and the no free lunch theorem." In Search Methodologies, pp. 317–339. Springer, Boston, MA, 2005.]], but leading econometricians argue that if their discipline is to remain relevant for students, a balance must be struck between *using data to solve problems*[12] while preserving the strengths of applied econometrics (Athey and Imbens 2019). Encouragingly, there have been recent advances in theoretical(Athey and Wager 2017; Wager and Athey 2017; Athey et al. 2019; Athey, Tibshirani, and Wager 2019) and causal(Zhao and Hastie 2021) properties of machine learning models published in high-quality economics and statistics journals (Zuo, Stewart, and Blume 2021; Apley and Zhu 2020).

2.6 Confronting statistical inference education in finance

Traditionally, econometrics has favoured the frequentist paradigm. However, with the increase in computer power, improvements in Markov Chain Monte Carlo (MCMC) methods, and advances in probabilistic programming, Bayesian inference is becoming more popular in the industry. However, teaching statistical inference in finance is still dominated by frequentism, with Bayesian inference a footnote at best. The perception among some educators is that Bayesian statistics are too complex a topic for an introductory course. However, the human brain is an inference engine that intuitively learns concepts that are more intuitive and easier to teach in an introductory course. A naive assumption is that these paradigms are competing, and in traditional econometrics, the frequentist approach is preferred due to its ability to be more

[12]This is framing econometrics as decision making under uncertainty(Dreze 1972; Chamberlain 2000, 2020)

objective. Nevertheless, objectivity is explicitly linked to the analyst's confidence in the prior belief that asymptotic properties of frequentist methods hold. However, Bayesian inference does require the additional setting to prior probabilities.

The boundary between econometrics and ML is much debate (Lommers, Harzli, and Kim 2021). However, in applied work, the reality is much more nuanced, with many methods falling into both camps. For instance, the bootstrap facilitates statistical inference and ensemble methods, such as the Random Forest algorithm. Classical econometrics requires a model that incorporates our knowledge of the economic system[13], and ML requires us to choose a predictive algorithm with reliable empirical capabilities. Justification for an inference model typically rests on whether we feel it adequately captures the essence of the system. Likewise, the choice of pattern-learning algorithms often depends on measures of past performance in similar scenarios. Thus, inference and ML can complement us to economically meaningful conclusions.

3 Teaching open science analytics in the digital age

his section outlines how computation has evolved in both finance and as a utility in the cloud. We then profile our computing use case as a central tenant of teaching open-source statistical inference in finance. Next, we detail the environment to focus on inference in the first 15 minutes of an econometrics course lecture one. We then discuss the course implementation and the toolbox used for success. Finally, we argue that this framework nurtures responsible research practices in statistical inferences that ultimately supercharge employability.

3.1 Brief history of computing in finance and the cloud

For centuries, finance and computation have gone hand in hand, with quantitative finance taking its roots from Bachelier's *Theory of Speculation* (Bachelier 1900). Computing as a utility can be traced back to Professor John McCarthy in the early 1960s. As computing power has become more accessible and affordable, computation has become a central finance part. Figure 1 illustrates some of the critical moments in the development of computing in finance and the cloud.

[13]The more popular frequentist paradigm depends on the behaviour of estimators under increasing sample size falls under the heading of "asymptotic theory." The properties of most estimators in the classical world can only be assessed "asymptotically," i.e. are only understood for the hypothetical case of an infinitely large sample. Also, virtually all specification tests used by frequentists hinge on asymptotic theory. This is a significant limitation when the data size is finite(Dixon and Polson 2020).

Figure 5: Computing landmarks finance and cloud computing. The data for the cloud computing timeline is sourced from Varghese et al. (2019), while the finance timeline is the authors' calculations

On the *buy-side*, in the early 1950a, Harry Markowitz transforms quantitative approaches to portfolio management. For example, Markowitz solved a complex mean-variance portfolio optimisation problem using algorithmic programming. Meanwhile, in the early 1960s Ed Thorp and John Simons, using computer-aided statistical algorithms, showed how arbitrage opportunities, unseen by traditional hedge fund managers, could be exploited to beat the market *consistently*.

On the *sell-side* a game-changing breakthrough in the 1970s was a model to price derivative products (Black and Scholes 1973; Merton 1973) (BSM model), resulting in the explosive growth of options markets (Cesa 2017). Subsequently, weaknesses in the BSM model fuelled growth in financial computing. Quantitative researchers, with the increased availability of computing power, used more realistic continuous-time pricing models to estimate complex partial differential equations (Reisinger and Wissmann 2018).

3.2 Teaching environment for computing

Much like teaching statistics and data science, embedding computing in a financial statistics course has three interconnected teaching advantages:

1. Produce interesting output with data (and code) within the first ten minutes of the first class; A have a knock-on effect of challenging students to infer meaning from data and statistics from day one;
2. Get students to think about computation as an integral part of the finance curriculum(Kaplan 2007; Çetinkaya-Rundel and Rundel 2018))
3. Demystify the folk theorem of statistical computing where students think that changing the computing environment improves their output;

A standard solution is to use computing labs to facilitate computation exercises. However, one downside to this approach is that instructors usually do not have administrative access and therefore struggle to accomplish basic maintenance tasks, such as pre-loading module-specific content. Furthermore, this usually leads to a familiar environment for all courses, rather than specialised setups for more advanced computational methods. Finally, the most significant downside is that using computing labs discourages active engagement of computation in all aspects of the module.

Our approach has been to use a browser-based cloud computing solution to provide a frictionless student experience in lectures and workshop sessions. Using the sizeable academic discount, we use the RStudio Teams enterprise software packages and manage student access using a container farm of dockerised instances. In addition, the Workbench software in the Teams suite (formerly RStudio server pro) allows online access to several integrated development environments (IDEs)[14] to script in both R and Python ("RStudio Workbench" 2021).

Compared to the computer labs approach, our approach has three distinct benefits:

The passive lecturing then active labs are replaced by dynamic lectures and labs and 24/7 access to computing for active independent learning; Help students who have cost constraints or limitations to accessing computing hardware; Ease of sharing code, data and environments.

3.3 Why R and Python?

R and Python are the two leading languages used in the industry for data analysis. Thus, to best prepare students to be competitive in the job market, we made the explicit decision to teach both languages at master level[15]. Although some notable holdouts teach econometrics using commercial graphical user interfaces(GUI), these languages have infiltrated academia. Proponents of GUI-based econometrics teaching argue that teaching statistical concepts is less intimidating to beginners when using a point-and-click approach than command line methods. Furthermore, the argument goes that teaching programming and statistics in tandem creates too much friction for students.

In our experience, such convenience is only possible by removing data analysis from the course content and providing students with tidy, rectangular data. However, this approach is a disservice to students for modern financial data analytics. Furthermore, point-and-click procedures require a bespoke student user manual that can run to 40-plus pages.

We argue there is a significant learning curve for the novice student, which is not generalisable to other analytics workflows. In general, using a GUI *copy and paste* workflow can increase student frictions, be more error-prone, be harder to debug, and, most importantly, disconnect the logical link between computing from financial analytics(Baumer et al. 2014). However, perhaps most important is that by learning generalisable coding/data skills, a student an adequately prepared to into an industry where technologies are rapidly evolving.

[14] To date, the software ships with a Launcher package that facilitates access to Jupyter notebooks, Jupyterlab, RStudio IDE, and Visual Studio

[15] At present, MSc in Quantitative Finance uses both languages, and we hope to expand this to all finance programmes and the new Actuarial Science masters in the future

3.4 Why RStudio Teams?

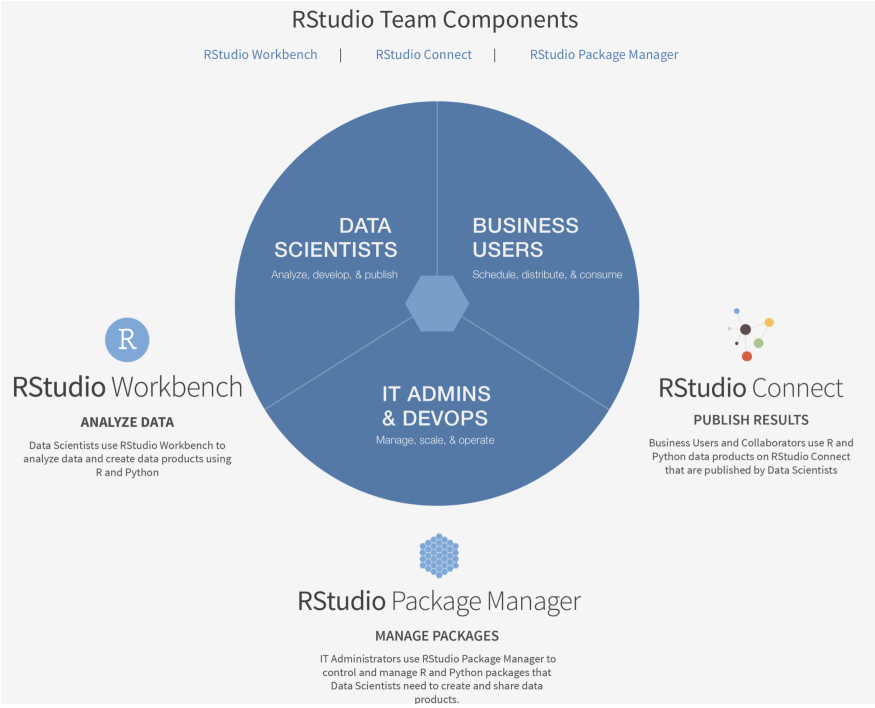

Figure 6: The three components of the RStudio Enterprise Team Bundle

Figure @ref(fig:rstudioteams) visualises the components that make up the RStudio Team bundle.

RStudio describes this product as follows:

> RStudio Team is a bundle of RStudio's enterprise-grade professional software for scaling data science analytical work across your team, sharing data science results with your key stakeholders, and managing R and Python packages. RStudio Team includes RStudio Workbench, RStudio Package Manager, and RStudio Connect. RStudio Team offers convenience, simplicity, and savings to organisations using R, Python and RStudio at scale.

- ("RStudio Team" 2021)

Teams is an enterprise-grade setup offered free of charge for academic teaching. This discount is a significant saving for educational budgets, typically between £10,000 to £15,000. The School's budget can then focus on purchasing an agile computing infrastructure.

For teaching computation, the IDE is the most critical tool in this bundle. The Workbench product comes with Jupyter (notebook and lab) and RStudio native IDE, which provide a powerful interface that helps flatten the learning curve in command line teaching. It has a series of panes to view data, files, and plots interactively. Additionally, since it is a full-fledged IDE, it also features integrated help, syntax highlighting, and context-aware tab completion.

Students access the RStudio IDE through a centralised RStudio server instance, which allows us to provide students with uniform computing environments. Furthermore, the IDE integrates directly with some critically essential tools for teaching best practices and reproducible research, such as R Markdown, Docker, and Git version control.

Importantly, we do not dissuade students from creating local instances of R and Python, but we do not want it to be a prerequisite of any module. Students are then allowed to progressively develop their setup to know that fully-fledged instances are always departmental resources.

3.5 Remote RStudio Workbench Platform

A popular approach to running a centralised RStudio server in teaching computation in higher-level statistics courses is to build a shared infrastructure with high powered computation power. This hardware is usually housed securely on-premises and managed by a dedicated IT team. For example, the Duke University statistics department purchased and operated a powerful farm of computer servers that can serve approximately 100 students per semester (Çetinkaya-Rundel and Rundel 2018). We have chosen to run RStudio Workbench using virtualised hardware on the Microsoft Azure cloud. Figure @(fig:current-setup) shows the architecture of the current setup (without dockerisation). Each student is assigned a Linux account, authenticated using a departmental login. Students then connect to a single RStudio Workbench instance, and via the Launcher, the software can open an IDE to access Python or R scripting environments. Thus, each student experiences a similar computing environment solving the perennial. **but it worked on my machine?** problem.

The primary advantage of running and managing a cloud computing platform is control. Lecturers control a shared user environment for each course, including required packages, resource configuration, remove or kill sessions and monitor resource demand on the system. This management work adds a considerable burden to the lecturer and the IT support, partially offset by the time saved supporting the build of lab-based PCs. However, our experience and student feedback suggest that the benefits far outweigh these additional costs. Furthermore, not providing students with such a resource is a disservice to their employability in the modern world of finance.

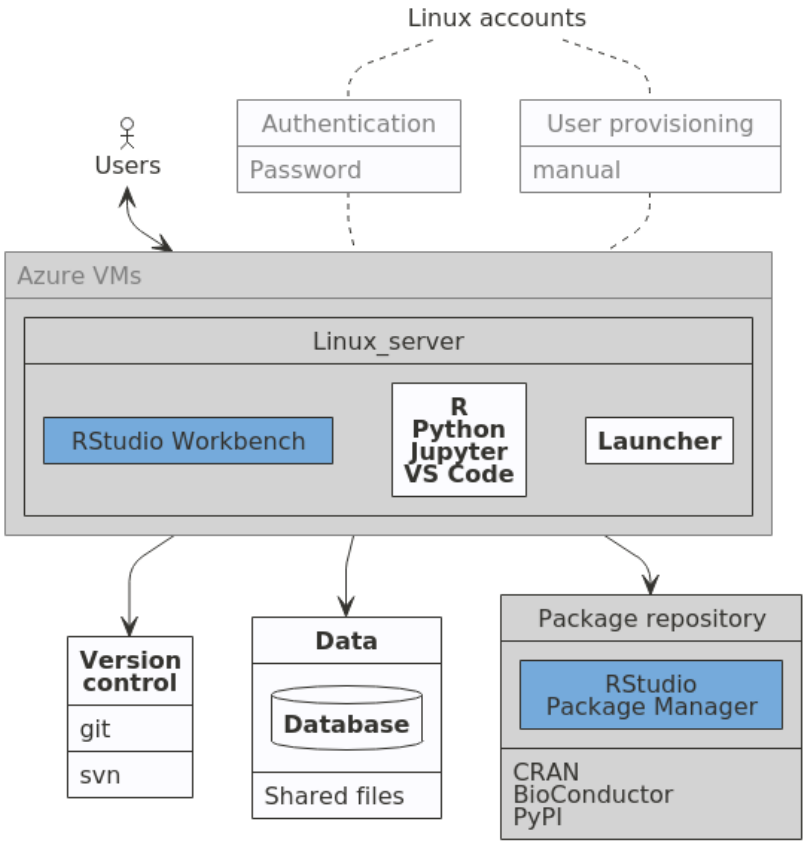

Figure 7: Current set up of RStudio workbench on Azure

3.6 Containerisation in finance

Linux containers are technologies that allow you to package and isolate applications with their entire runtime environment (International Banker 2017). Their strategic advantage is their application independence from

the underlying operating environment enabling standardisation and automation, significantly lowering cost and operational risk.

Virtualisation technology is the underlying cloud computing element, and containers take this to the next level. Cloud computing has traditionally used virtual machines to distribute available resources and provide isolated environments among users. The key difference between virtual machines and containers is that containers share the same underlying operating system (Mavridis and Karatza 2019)

Containerisation is decades old, but the emergence of the open-source Docker Engine has accelerated the adoption of this technology. Docker is a *lightweight* virtualisation technology that allows sharing one operating system so that all code, runtimes, tools, and libraries needed for a piece of software are made available. This *build once run anywhere* property makes them highly portable, agile and efficient approach to running **sandboxed** instances of RStudio Workbench. The open-source nature of Docker makes it a transparent and powerful tool for reproducible computational finance research. From a teaching perspective, each student can be mapped to a single container, secluding individual operates and maintaining strict control of computing resource usages to provide accidental disruption of individual students' work.

Furthermore, clusters can be deployed using a container orchestration system such as Kubernetes, and the operational overhead can be largely automated using AKS. Given that they are much lighter weight than VMs, a large container farm of RStudio instances can be run concurrently on one single server. We plan to build this infrastructure into our platform and have sketched out the planned setup in figure @ref(fig:future-setup).

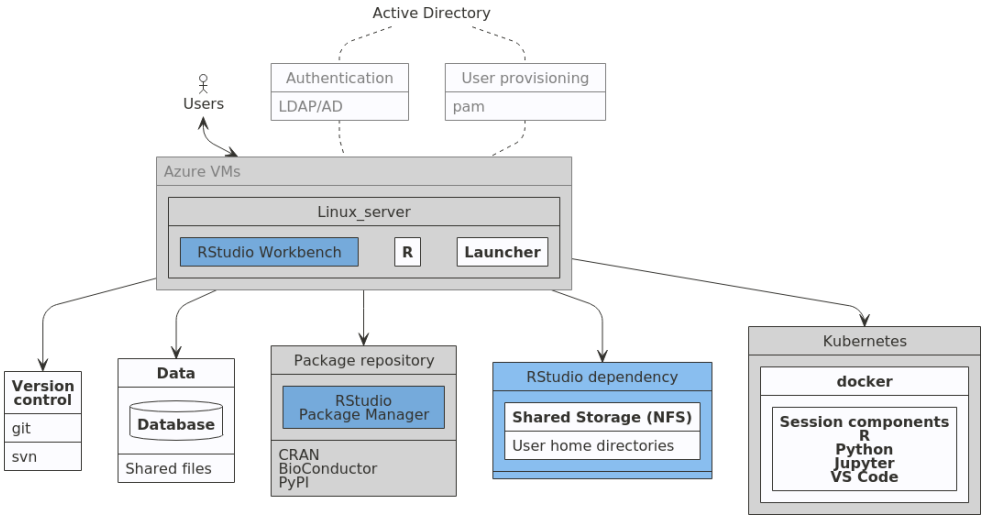

Figure 8: Dockerised set up of RStudio workbench on Azure

3.7 Course implementation

We piloted our new infrastructure at masters level teaching in the 2020-2021 academic year at Queen's Management School. Named Q-RaP (Queen's management school Remote analytics Platform), students used the platform in two modules; algorithmic trading and investment and time-series financial econometrics. Anecdotally, it received excellent feedback from students, especially when remote teaching and learning was the norm. In 2021/2022, it will be used in a further two masters level courses (research methods in finance and computational methods in finance) and available for some business analytics modules. As well as the teaching advantages, the resource has the additional benefit of easing the demand pressures on computer labs.

3.7.1 Reproducibility with computational notebooks

Computational notebooks are documents that combine code, discussion and output in a dynamic reproducible format. An essential advantage of computational notebooks is that they embody the PPDAC credible analysis workflow (Problem, Plan, Data, Analysis, Communication). PPDAC is the professional standard for data analysis and plausible inference(Spiegelhalter 2019). Unlike the copy and paste approach, all five parts of the

PPDAC approach can be included in one document, providing an enhanced level of transparency, portability and reproducibility.

There are two main formats for producing computational notebooks; Jupyter notebooks and R Markdown. Both are based on Markdown, one of the most popular markup languages. Using Markdown is different from using a WYSIWYG editor. In an application like Microsoft Word, you click buttons to format words and phrases, and the changes are visible immediately. In contrast, when creating a Markdown-formatted file, you add Markdown syntax to the text to indicate which words and phrases should look different. Markdown is highly portable, platform-independent, future proof, and essential for the modern financial data scientist.

Out of the box, the Jupyter ecosystem supports python scripting using the IPython kernel but can support up to 100 different languages (called 'kernels') by installing additional kernels[16]. Jupyter notebooks are a lightweight, low learning curve approach to teaching computing and are an excellent way to get non-technical students up and running in the first 10 minutes of a course. R Markdown is probably one of the most powerful tools in the RStudio IDE. R Markdown files are plain text documents that combine text, code and YAML metadata into an authoring framework for financial analytics. In the RStudio IDE, you can open an. Rmd file and working interactively, or render the file to build a static report or a dynamic web app using the `Shiny` packages. For instance, when you render an R Markdown document, it will combine the text with output from your code. The rendering process produces static formats such as HTML, pdf and word. However, it can also produce interactive dashboards, web apps, slide shows, websites, and technical documentation (See video below). We mainly use Python and R code chunks in our teaching, the former output in the RStudio environment using the `reticulate` package.

Pedagogically, the main benefit of R Markdown and Jupyter notebooks is to embed the logical connection between computing and financial data analysis. This approach is sometimes referred to as *literate programming* (Knuth 1984)[17], which made code, output and narrative inseparable. Computational notebooks have four advantages over the copy-and-paste approach:

1. Combining code and output in one document makes it easier for a student to locate the source of the errors and encourages more experimentation;
2. Strict uniformity of the reporting template makes it easier for the lecturers to grade;
3. Collaboration and group projects become easier for students when using version control. Version control also provides a strict tagging system of individual contribution is assessed within a group work setting;
4. Provides a baseline template document that, as students learn, can be more and more lightweight.
    - By removing the scaffolding in a slow, piecemeal way as the course progresses, active learning appeals.

On balance, using *literate programming* via computational notebooks has meaningful learning and employability benefits, especially as it is becoming a standard approach to collaboration in the finance industry.

3.7.2 Version control, git and GitHub

Increasingly, in the world of computational finance, version control is being used to disseminate and promote innovative coding solutions to financial problems. Furthermore, in line with applied statistics curricula (Çetinkaya-Rundel and Rundel 2018), modern finance curricula should strive to have students produce reproducible output. Git is a popular command-line version control tool that integrates well with RStudio Teams. In addition, GitHub is a web-based hosting repository platform that provides access control and many more collaborative features to manage teamwork on computing projects.

From a finance industry employability perspective, in the past, there has been considerable resistance to the user of externally hosted IT services as security is paramount to highly regulated financial institutions. The opposition has typically been for strategic and economic reasons:

- For companies that have swallowed the Windows *Koolaid* there are more secure options such as Mercurial

[16] `https://jupyter4edu.github.io/jupyter-edu-book/jupyter.html`

[17] Donald Knuth is pioneering in the computing world and creates the vastly popular TeX typesetting markup language

- It is cheaper for large companies to do it in house
- In a large organisation, there are guaranteed to be fiefs all wanting to do things their way. So a standardised version control system is the only appeal for an obvious Total Cost of Ownership benefits.

These arguments are outdated, especially with Big Tech acquisition activity in the git ecosystem space. For example, in 2018, Microsoft bought GitHub and soon after Alphabet's Google Ventures took a significant stake in GitLab. This has propelled git version control as an industry standard that is now easily integrated into all legacy systems, including Windows Servers.

Students are required to use git for all assignments in the classroom, where GitHub is a central repository where students can upload their work and provide feedback. Recently GitHub Classroom was introduced, providing an enterprise-level service free of charge for academic teaching.

Before GitHub classrooms, GitHub management tools such as organisation and teams can be set up privately. Only the students or the group can see and contribute to the assignment. For example, we used a model where each module has a separate organisation to which students are invited at the beginning of the semester. The teams' tool allows creating a separate team-based repository with finer-grained access control for group work. In addition, the instructor can monitor each student's progress and contribution with administrative access through the continuous integration functionality. GitHub classroom provides automated instant feedback on simple process tasks, for example, checking for common reproducibility mistakes in R Markdown submissions. Feedback on larger prediction projects can be automated using instant accuracy scores and live leader-boards similar to a Kaggle contest (Çetinkaya-Rundel and Rundel 2018).

Much of what has been described above has now been automated in GitHub Classroom and can also be integrated into learning management systems such as Canvas. The learning curve for these tools is unavoidable. It can be high for introductory-level courses, but a basic understanding of the workflow in Figure 5 is sufficient for most modules.

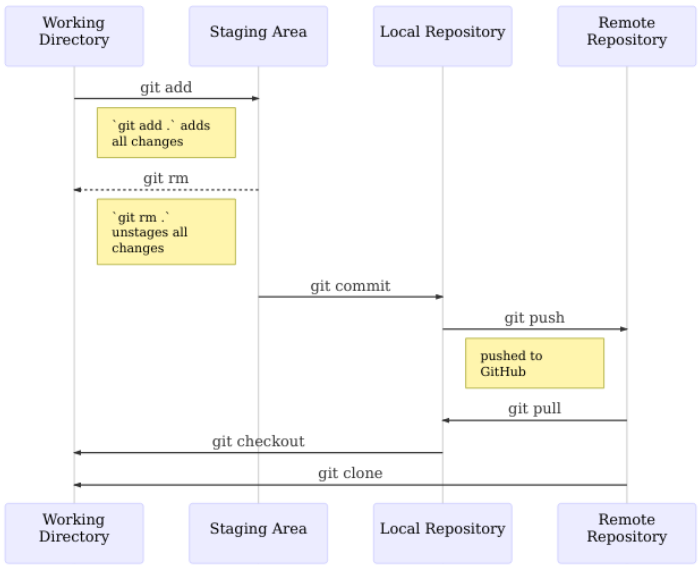

Figure 9: Seven git commands students need to learn

4 Discussion and concluding remarks

Our primary objective is to foster industry-ready graduates for the fast-paced digital age as finance educators. As academics, we need to embrace the open science movement to remain credible and relevant as educators and scientists. As we enter a new phase in the development cycle of financial technology, exposing students to industry-standard computing technologies is a good start. Our goal is to reduce the frictions in teaching computation and statistical inference in finance. Our vision is to expand this platform to all the management school's quantitative modules.

Pedagogically, by embedding computation in a centralised frictionless way, we can spend more time developing the essential communications skills for explaining the *why* of the output from the code and data. Teaching econometrics and statistics in business schools is a considerable challenge, especially with students from non-technical backgrounds. The traditional approach mathematical formula first-application after only disenfranchises students from statistical computing further and is a disservice to the modern business school graduate. We find the learning curve is significantly flattened by a code-first approach, increasing student buy-in with approachability and usability. In addition, mathematical formulas can be introduced to build a deeper understanding of statistical plumbing and critical thinking around limitations. Finally, using open-source software, strict version control, and reproducible notebooks embeds the principles of open science in the curricula.

The infrastructure and toolkit we described above ensure buy-in by making computing a central component of courses and assessments. Using GitHub as the sole course management tool forces students to become familiar early, ensuring questions and problems are dealt with at least before the first assignment date. Furthermore, requiring students to submit assignments using R Markdown forces students to use a literate programming approach, ensures reproducibility and embed the PPDAC principles in their work. Finally, indoctrinating students early with these reproducibility principles inoculates any bad computational habits forming.

Importantly, we want to enable students and colleagues to centralise computation in frictionless and agile education. We hope this can result in a more meaningful approach to *solving business problems with data* in a more thoughtful, transparent and credible manner. Nevertheless, perhaps most important is that by learning generalisable coding/data skills, a student an adequately prepared for an industry where technologies are rapidly evolving.

References

Apley, Daniel W, and Jingyu Zhu. 2020. "Visualizing the Effects of Predictor Variables in Black Box Supervised Learning Models." *J. R. Stat. Soc. Series B Stat. Methodol.* 82 (4): 1059–86.

Athey, Susan, Mohsen Bayati, Guido Imbens, and Zhaonan Qu. 2019. "Ensemble Methods for Causal Effects in Panel Data Settings," March. `https://arxiv.org/abs/1903.10079`.

Athey, Susan, and Guido W Imbens. 2019. "Machine Learning Methods That Economists Should Know About." *Annu. Rev. Econom.* 11 (1): 685–725.

Athey, Susan, Julie Tibshirani, and Stefan Wager. 2019. "Generalized Random Forests." *Aos* 47 (2): 1148–78.

Athey, Susan, and Stefan Wager. 2017. "Policy Learning with Observational Data," February. `https://arxiv.org/abs/1702.02896`.

Bachelier, Louis. 1900. "Theory of Speculation in the Random Character of Stock Market Prices." *MIT Press, Cambridge, Mass. Blattberg* 1018: 17–78.

Barredo Arrieta, Alejandro, Natalia Diaz-Rodriguez, Javier Del Ser, Adrien Bennetot, Siham Tabik, Alberto Barbado, Salvador Garcia, et al. 2020. "Explainable Artificial Intelligence (XAI): Concepts, Taxonomies, Opportunities and Challenges Toward Responsible AI." *Inf. Fusion* 58 (June): 82–115.

Baumer, B, Mine Çetinkaya-Rundel, Andrew Bray, Linda Loi, and N Horton. 2014. "R Markdown: Integrating a Reproducible Analysis Tool into Introductory Statistics." *Undefined*.

Black, Fischer, and Myron Scholes. 1973. "The Pricing of Options and Corporate Liabilities." *J. Polit. Econ.* 81 (3): 637–54.

Blume, Jeffrey D, Robert A Greevy, Valerie F Welty, Jeffrey R Smith, and William D Dupont. 2019. "An Introduction to Second-Generation p-Values." *Am. Stat.* 73 (sup1): 157–67.

Breiman, Leo. 2001. "Statistical Modeling: The Two Cultures (with Comments and a Rejoinder by the Author)." *Statistical Science* 16 (3): 199–231.

Cesa, Mauro. 2017. "A Brief History of Quantitative Finance." *Probability, Uncertainty and Quantitative Risk* 2 (1): 1–16.

Çetinkaya-Rundel, Mine, and Colin Rundel. 2018. "Infrastructure and Tools for Teaching Computing Throughout the Statistical Curriculum." *Am. Stat.* 72 (1): 58–65.

Chamberlain, Gary. 2000. "Econometrics and Decision Theory." *J. Econom.* 95 (2): 255–83.

———. 2020. "Robust Decision Theory and Econometrics." *Annu. Rev. Econom.* 12 (1): 239–71.

Cunningham, Scott. 2021. *Causal Inference: The Mixtape*. Yale University Press.

Dixon, Matthew F, and Nicholas G Polson. 2020. "Deep Fundamental Factor Models," March. `https://arxiv.org/abs/1903.07677`.

Dreze, Jacques H. 1972. "Econometrics and Decision Theory." *Econometrica*.

Easley, David, Marcos López de Prado, Maureen O'Hara, and Zhibai Zhang. 2020. "Microstructure in the Machine Age." *Rev. Financ. Stud.*, July.

Efron, Bradley, and Trevor Hastie. 2016. *Computer Age Statistical Inference*. Cambridge University Press.

Fisher, R A. 1936. "Design of Experiments." *Br. Med. J.* 1 (3923): 554–54.

Gelman, Andrew, Jennifer Hill, and Aki Vehtari. 2020. *Regression and Other Stories*. Cambridge University Press.

Haavelmo, Trygve. 1943. "The Statistical Implications of a System of Simultaneous Equations." *Econometrica* 11 (1): 1–12.

Hang, Bai, Kewei Hou, Howard Kung, Erica X N Li, and Lu Zhang. 2019. "The CAPM strikes back? An equilibrium model with disasters." *J. Financ. Econ.* 131 (2): 269–98. `https://doi.org/10.1016/j.jfineco.2018.08.009`.

Hastie, Trevor, Robert Tibshirani, and Jerome Friedman. 2009. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction, Second Edition*. Springer Science & Business Media.

International Banker. 2017. "The Benefits of Leveraging Containers in the Financial Services Industry." `https://internationalbanker.com/technology/benefits-leveraging-containers-financial-services-industry/`.

Investment Banking Council. 2020. "AI in Investment Banking - the New Frontier." `https://www.investmentbankingcouncil.org/blog/ai-in-investment-banking-the-new-frontier`.

Jaeger, Markus, Stephan Krügel, Dimitri Marinelli, Jochen Papenbrock, and Peter Schwendner. 2021. "Interpretable Machine Learning for Diversified Portfolio Construction." *The Journal of Financial Data Science*, June, jfds.2021.1.066.

Jensen, Theis Ingerslev, Bryan T. Kelly, and Lasse Heje Pedersen. 2021. "Is There a Replication Crisis in Finance?" *SSRN Electronic Journal*. `https://doi.org/10.2139/ssrn.3774514`.

Kaplan, Daniel. 2007. "Computing and Introductory Statistics." *Technology Innovations in Statistics Education* 1 (1).

Knuth, D E. 1984. "Literate Programming." *Comput. J.* 27 (2): 97–111.

L, Ronald, Wasserstein, and Nicole A Lazar. 2016. "The ASA's Statement on p-Values: Context, Process, and Purpose." *Am. Stat.* 70 (2): 129–33. `https://doi.org/10.1080/00031305.2016.1154108`.

Lin, Sin-Jin, and Ming-Fu Hsu. 2017. "Incorporated Risk Metrics and Hybrid AI Techniques for Risk Management." *Neural Comput. Appl.* 28 (11): 3477–89.

Lommers, Kristof, Ouns El Harzli, and Jack Kim. 2021. "Confronting Machine Learning with Financial Research." *The Journal of Financial Data Science*, June, jfds.2021.1.068.

López de Prado, Marcos. 2018. *Advances in Financial Machine Learning*. John Wiley & Sons.

———. 2019. "A Data Science Solution to the Multiple-Testing Crisis in Financial Research." *The Journal of Financial Data Science* 1 (1): 99–110.

Mahdavi, Mehrzad, and Hossein Kazemi. 2020. "It's All about Data: How to Make Good Decisions in a World Awash with Information." *The Journal of Financial Data Science* 2 (2): 8–16.

Mavridis, Ilias, and Helen Karatza. 2019. "Combining Containers and Virtual Machines to Enhance Isolation and Extend Functionality on Cloud Computing." *Future Gener. Comput. Syst.* 94 (May): 674–96.

Merton, Robert C. 1973. "Theory of Rational Option Pricing." *The Bell Journal of Economics and Management Science* 4 (1): 141–83.

Molina, Mario, and Filiz Garip. 2019. "Machine Learning for Sociology." *Annu. Rev. Sociol.* 45 (1): 27–45.

Pearl, Judea. 2009. *Causality*. Cambridge University Press.

R, Campbell, Harvey. 2017. "Presidential Address: The Scientific Outlook in Financial Economics." *SSRN Electronic Journal*, July. https://doi.org/10.2139/ssrn.2893930.

Reisinger, Christoph, and Rasmus Wissmann. 2018. "Finite Difference Methods for Medium-and High-Dimensional Derivative Pricing PDEs." In *High-Performance Computing in Finance*, 175–95. Chapman; Hall/CRC.

Responsible Research in Business & Management, Community of. 2020. "A Vision of Responsible Research in Business and Management Striving for Useful and Credible Knowledge." https://rrbm.network/wp-content/uploads/2020/04/Position-Paper_revised_8April2020.pdf.

"RStudio Team." 2021. https://www.rstudio.com/products/team/.

"RStudio Workbench." 2021. https://www.rstudio.com/products/workbench/.

Rubin, Donald B. 1974. "Estimating Causal Effects of Treatments in Randomized and Nonrandomized Studies." *J. Educ. Psychol.* 66 (5): 688–701.

Society of Actuaries. 2020. "The Powerful Combination of Actuarial Expertise and InsurTech Knowledge." Society of Actuaries; https://www.soa.org/programs/insurtech/.

Spiegelhalter, David. 2019. *The Art of Statistics: Learning from Data*. Penguin UK.

Wager, Stefan, and Susan Athey. 2017. "Estimation and Inference of Heterogeneous Treatment Effects Using Random Forests." *J. Am. Stat. Assoc.*, April, 1–15.

Wolpert, D H, and W G Macready. 1997. "No Free Lunch Theorems for Optimization." *IEEE Trans. Evol. Comput.* 1 (1): 67–82.

Wright, Sewall. 1934. "The Method of Path Coefficients." *Aoms* 5 (3): 161–215.

Zhao, Qingyuan, and Trevor Hastie. 2021. "Causal Interpretations of Black-Box Models." *Journal of Business & Economic Statistics* 39 (1): 1–19. https://doi.org/10.1080/07350015.2019.1624293.

Zuo, Yi, Thomas G Stewart, and Jeffrey D Blume. 2021. "Variable Selection with Second-Generation p-Values." *Am. Stat.*, June, 1–21.