Machine Learning on Tactical Asset Allocation with Machine Learning and MATLAB Distributed Computing Server on Microsoft Azure Cloud

Emilio Llorente-Cano

James Mann

Aberdeen Asset Management, Plc

Contents

Aberdeen

Purpose of this presentation

- What are the key elements in such an Investment Process:
  - The product profile (TAA, SAA, Absolute Return, Smart Beta)
  - The factors that define our market regimes
  - The question to the Machine
  - How the various algorithms identify the associations between regimes and assets performance
  - High Performance Computing
- Is it that simple? Challenges and rewards

Why Machine Learning

The problem

- Financial time series can be separated by inputs (factor drivers) and outputs (assets performance). The cause-effect relationships between them are non-deterministic, non-linear and multidimensional
- Professional investor's task: Discover the relationships between economic events and market performance

The Solution

- Machine Learning ability: Discover nonlinear, previously unknown, associative structures within complex datasets
- The relationships between market inputs and outputs described by Machine Learning can then be seen as a behavioural pattern, proper for investment decisions: Buy/Sell, Long/Short, Overweight/Underweight.

Adaptive Machine Learning Investment Process

STEP 1: Client, Product

- Expected performance
- Accepted tolerance to core risk measures
- Allocation constrains

STEP 2: Investable Investment Universe

- Time horizon
- Trends and returns
- Correlations

STEP 3: Market regimes

- Drivers, Factors
- Noise, Signal
- Unbiased, Logic foundations

High Performance Computing

STEP 4: Machine Learning decision

- Structure + Out of Sample test = validation
- Best allocation match under current regime

Machine Learning Investment Committee

- **kNN**
- **CaRT**
- **SVM**
- **NNs**

STEP 5: Probability based allocation

- ML output = Class + Probability
- Used to construct the bespoke Product

Example on a TAA against Benchmark

55% Global Equities

40% DM

60% EM

60% Emerging Markets

45% EM Asia | 25% EM Latam | 30% EM Europe

STEP 6: Management

- Events monitored by the algorithms
- Performance and visualisation

Aberdeen

Adaptive Machine Learning Investment Process

- The framework generates asset allocation decisions on traditional and alternative asset classes: equities, rates, credit, currencies, commodities and smart beta strategies for various factors (value, growth, carry, volatility, momentum).

Market factors definition	Machine learning input	Question to the machine
Macroeconomic; Value; Liquidity; Sentiment	Data cleansing, signal processing and transformation in order to define the input to the Machine Learning algorithms	• Tactical Asset Allocation: underweight FTSE 100 to overweight S&P 500 • Systematic Global Macro: Short FTSE 100 **vs** Long S&P 500 • Tactical Smart Beta: underweight MSCI Value to overweight MSCI Growth Index
Leveraging industry and academic research for factor pre-selection	Data transformation for interpretability	Defining the problem

- Several Machine Learners outputs are combined using advanced techniques to create one final coherent asset allocation and the corresponding trades.
- The use of high performance computing is a key technology for backtesting and live trading

Contents

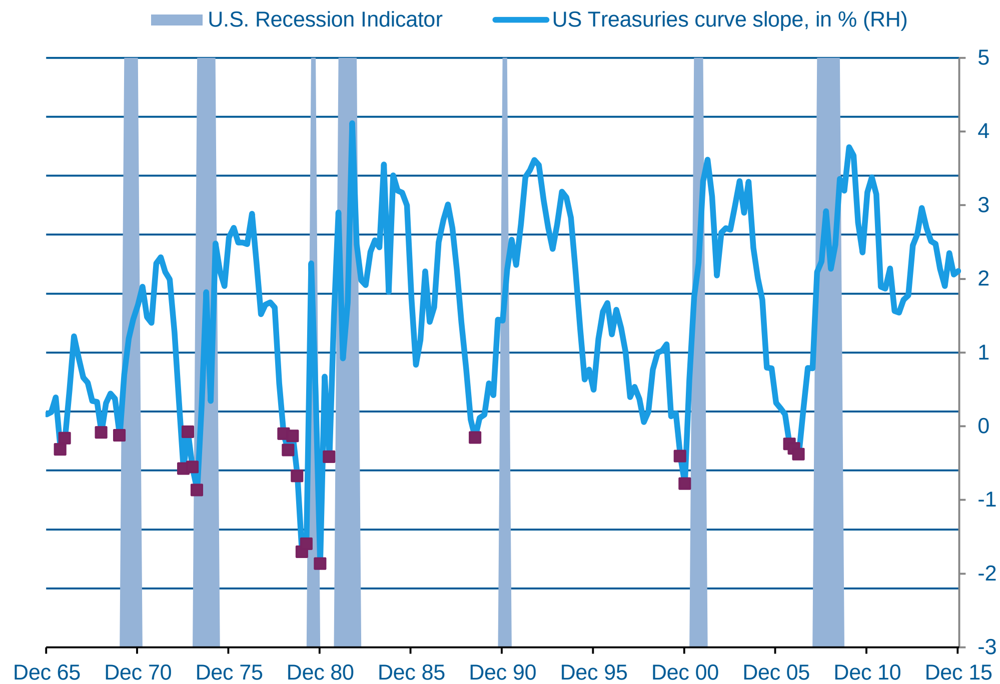
- Motivation
- **Market Factors & Data Interpretability**
- Question to Machine & Learning Algorithms
- Using MATLAB Distributed Computing Server (MDCS) cluster on Azure
- Q & A

Market Factors

- Factors are defined as explanatory variables that can drive markets performance.

- These factors must be:
  - <u>Comprehensive</u>: Include all necessary elements to identify a market regime.
  - <u>Explainable</u>: Based on solid research foundations.
  - <u>Persistent</u>: Their presence and influence is observed during different market cycles.
  - <u>Accessible</u>: Easily available to make the process reactive to the most updated information and maintain its continuity.

- Quoted factors: they include market participants expectations about future market events.



Source: Bloomberg

"The 3m vs 10y T-Bill spread is a valuable forecasting tool, that significantly outperforms other macroeconomic indicators in predicting recessions two to six quarters ahead."

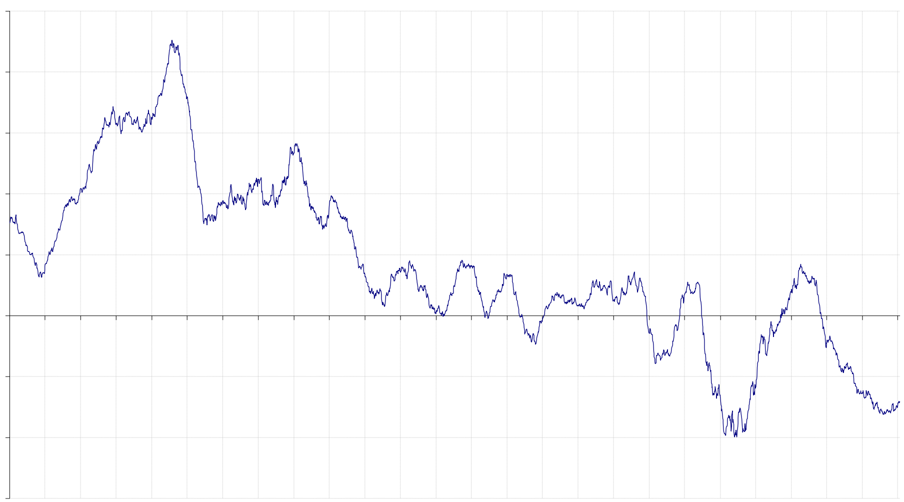
–Arturo Estrella and Frederic S. Mishkin, Federal Reserve Bank of New York

Data Interpretability

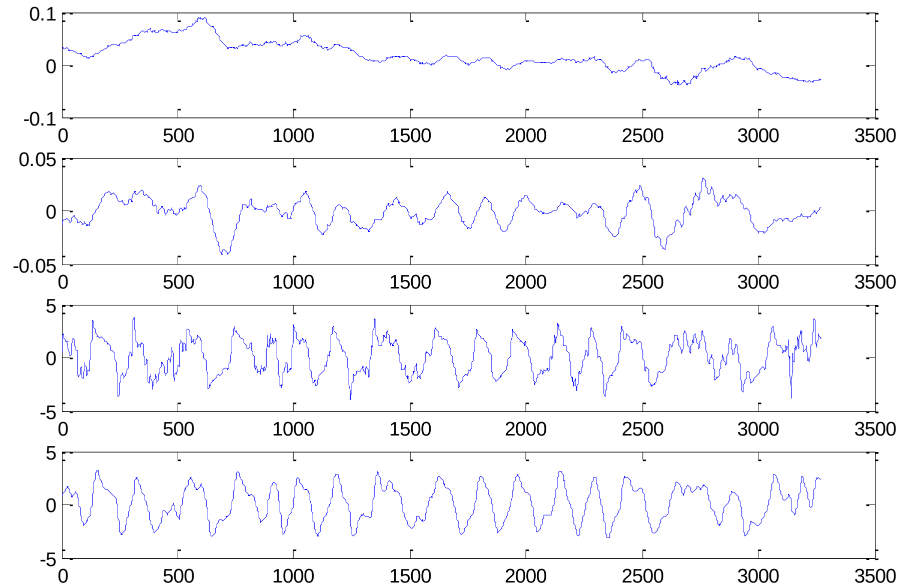
- Factors could be transformed using standardisation , clustering techniques that allow all algorithms to use as inputs the same information.
- These transformations allow for a better interpretability by the machine.
- Wavelet decomposition to reduce the risk of learning misguided by noise, with little loss of information.

Example: Money Flows into equity markets as a percentage of the country's index total capitalization.

Original time series



Transformed time series: from Z-Score to Wavelet denoising

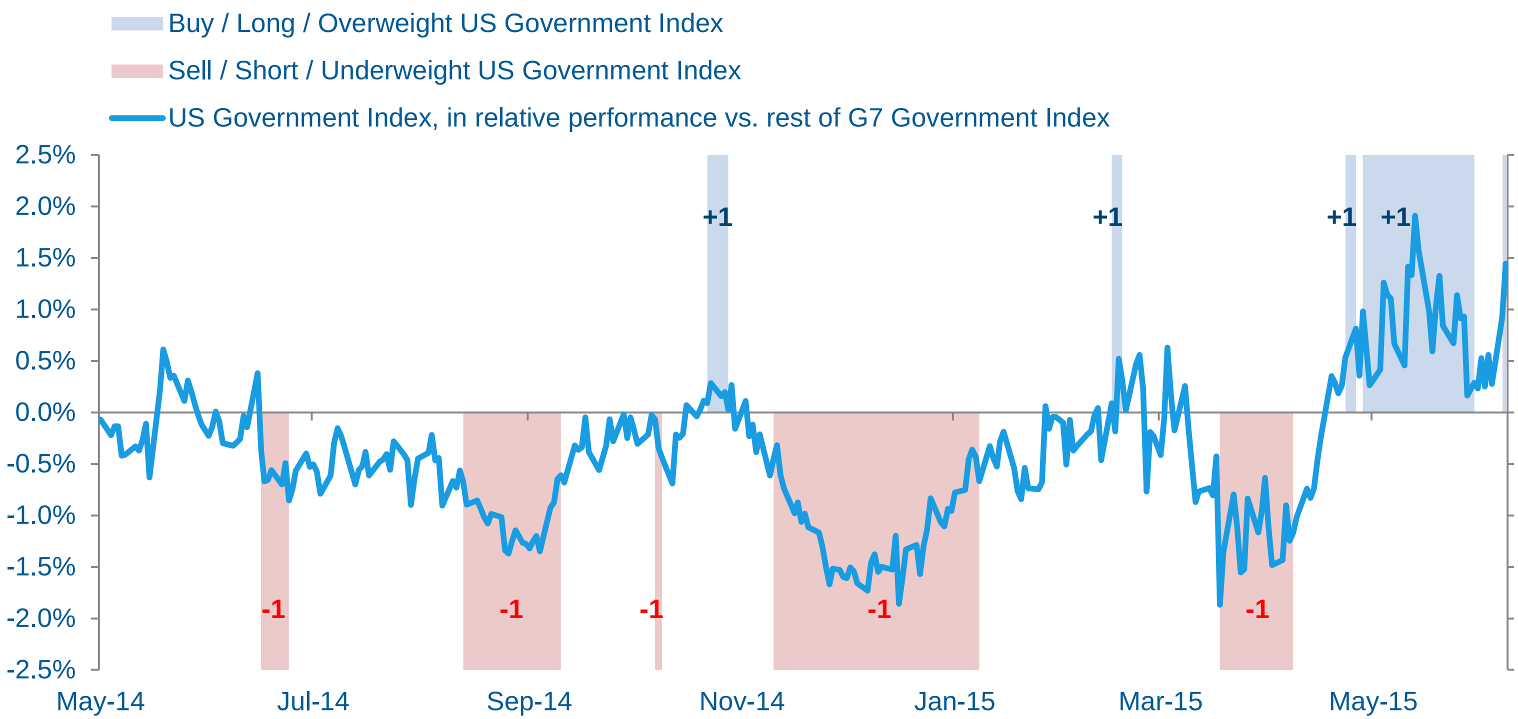


Contents

Aberdeen

Question to the Machine

- We focus on relative and absolute asset class performance **trends** over a defined time horizon, that is adapted according to the nature of the investment solution
- This way, our problem is a classification one, which allows us to use machine learning algorithms proved successful in the Artificial Intelligence literature
- The number of trades we generate depends purely on the Client's/Product profile and return targets



Source: Bloomberg

Question to the Machine

- Definition of the problem is essential for the choice of the ML algorithm.
- Multi-asset Funds are an example of a multi-class decision (many asset classes!). Due to the correlation among these classes, traditional single-label classification methods are not directly applicable.
- When observing the trends of different asset classes, each instance is associated with multiple labels and the classes are not mutually exclusive but may overlap. In traditional multi-class (i.e., single label) an instance is only associated with a single label and, therefore, the classes are mutually exclusive.

MULTIPLE CLASSES

BINARY LABELS

From	Till	US Government Bonds	UK Government Bonds	Japan Government Bonds	Core-Euro Government Bonds	Periphery-Euro Government Bonds	Australian Government Bonds
01/04/2015	01/05/2015	Underperform	Underperform	Outperform	Outperform	Underperform	Underperform
02/04/2015	04/05/2015	Underperform	Underperform	Outperform	Outperform	Underperform	Underperform
03/04/2015	05/05/2015	Underperform	Underperform	Outperform	Outperform	Underperform	Underperform
06/04/2015	06/05/2015	Underperform	Underperform	Outperform	Outperform	Outperform	Underperform
07/04/2015	07/05/2015	Underperform	Underperform	Outperform	Outperform	Outperform	Underperform
08/04/2015	08/05/2015	Underperform	Underperform	Outperform	Outperform	Outperform	Underperform
09/04/2015	11/05/2015	Underperform	Underperform	Outperform	Outperform	Outperform	Underperform
10/04/2015	12/05/2015	Underperform	Underperform	Outperform	Outperform	Outperform	Underperform
13/04/2015	13/05/2015	Underperform	Underperform	Outperform	Outperform	Outperform	Underperform
14/04/2015	14/05/2015	Underperform	Underperform	Outperform	Outperform	Outperform	Underperform
15/04/2015	15/05/2015	Underperform	Underperform	Outperform	Outperform	Outperform	Underperform
16/04/2015	18/05/2015	Underperform	Underperform	Underperform	Outperform	Outperform	Underperform
17/04/2015	19/05/2015	Underperform	Underperform	Underperform	Outperform	Outperform	Underperform
20/04/2015	20/05/2015	Underperform	Underperform	Underperform	Outperform	Outperform	Underperform
21/04/2015	21/05/2015	Underperform	Underperform	Underperform	Outperform	Outperform	Underperform

Source: Bloomberg

Learning Algorithms

- $X = Rd$ denotes the d-dimensional instance space,
- $\mathcal{Y} = \{y_1, y_2, \cdots, y_q\}$ denotes the label space with q possible class labels.
- Task of multi-label learning: to learn a function $h : X \rightarrow 2^{Y}$ from the multi-label training set $D = \{(\mathbf{x}_i, Y_i) | 1 \leq i \leq m\}$. For each multi-label example $(\mathbf{x}_i, Y_i)$, $\mathbf{x}_i$ is a d-dimensional feature and $Y_i \subseteq \mathcal{Y}$ is the set of labels associated with $\mathbf{x}_i$.
- Traditional machine learning algorithms adapted: k-Nearest Neighbours (kNNs), Decision Trees, Support Vector Machines (SVM), Neural Networks.
- Classification results:
  - Trade **Direction**: for any unseen instance $\mathbf{x} \in X$, the multi-label classifier $h(\cdot)$ predicts $h(\mathbf{x}) \subseteq \mathcal{Y}$ as the **set of proper labels** for **x**.
  - Trade **Weight**: simultaneously, the answer returned by a multi-label learning system corresponds to a **real-valued function** $f : X \times \mathcal{Y} \rightarrow R$, where $f(\mathbf{x}, y)$ can be regarded as the confidence of $y \in \mathcal{Y}$ being the associated label of **x**.

Aberdeen

(R)evolution

Now

Portfolios
- Tactical asset allocation
- Systematic Global Macro
- Tactical smart beta

Machine Learning
- Multi-label
- Multi-class

AI Algorithms
- Ensemble methods
- Support Vector Machines
- Neural Networks

2011 LIBSVM
R2011a Boost
R2013b Neural Networks

Data Management
- Clustering
- Wavelet filtering
- Market implied
- Market news

R2013b Datafeed toolbox Bloomberg history

Programming
- Object Oriented
- Cloud computing

R2008b OOP
R2012b MDCS

Before

Portfolios
- Risk on / Risk off
- Equities / Bonds / Cash
- Return vs Risk

Machine Learning
- Binary
- Ternary

AI Algorithms
- Classification Trees
- Support Vector Machines

R2009a Treebagger
R2013a SVM

Data Management
- Market priced
- Macroeconomic surveys

R2007b Spreadsheet link

Programming
- Scripting
- Local cores

Computationally intensive ML algos

- The level of computing challenge is high. This is largely due to the jump in the volume of data being handled and the dimensionality / uncertainty involved in the analysis of a potentially-wide range of assets classes; each of which will require careful pre and post-processing to ensure the correct inputs for the algorithms.

- The inevitable level of uncertainty must be addressed and a robust statistical framework delivered around data selection and performance – something that is critical to the delivery of a tailored market view which will put a strain on the validation processes.

- All these need of an iterative back-test methodology highly time consuming for the computer.

- Support Vector Machines are a classic example of slow training, due to the quadratic programming problem they have to solve, with the number of variables equal to the number of training data.

- 5 years ago, our first implementation of the whole backtest process for this algorithm took 24 hours of computer time. Today, the picture is very different.

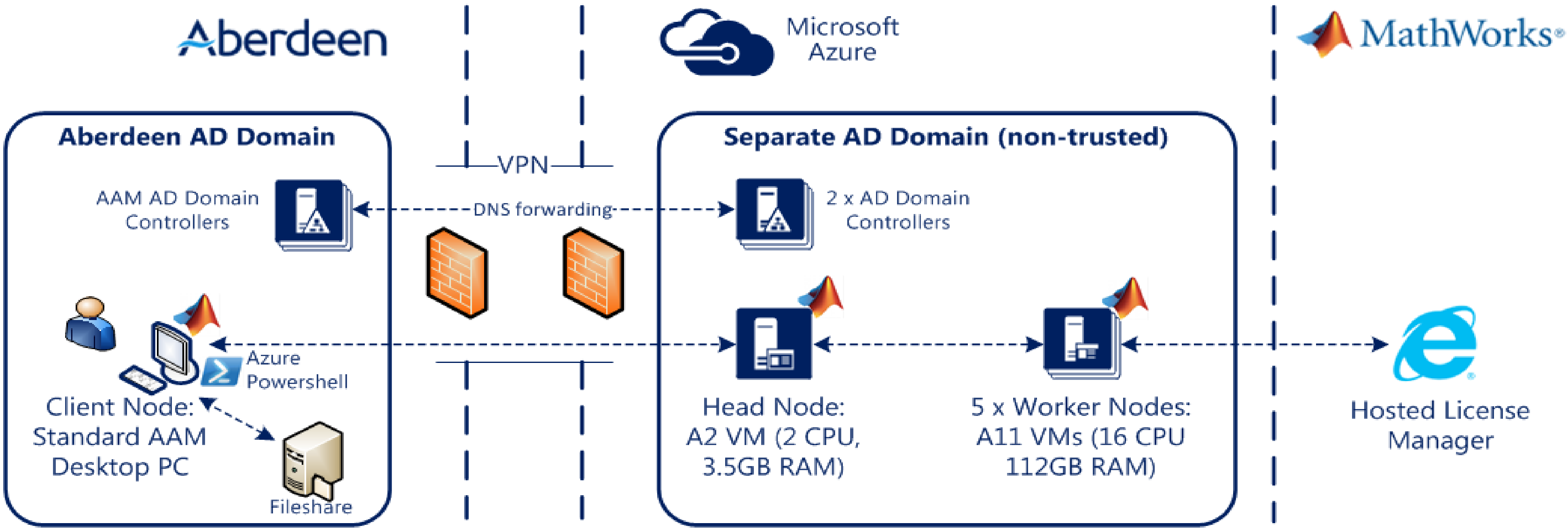
Contents

- Motivation
- Market Factors & Data Interpretability
- Question to Machine & Learning Algorithms
- **Using MATLAB Distributed Computing Server (MDCS) cluster on Azure**
- Q & A

The Challenge

- Since 2013 Emilio's team used an MDCS cluster hosted on-premise in Scottish Widows Investment Partnership (SWIP)
- SWIP was then acquired by AAM and Emilio's team were one of the first to migrate over to AAM platforms mid-2014
- An equivalent MDCS cluster capability had to be created in AAM as a prerequisite for Emilio's team migrating over to AAM
- The technology teams had about 4 weeks notice to achieve this
- Lead times for procuring and installing new hardware on-premise upon which to install the MDCS cluster were prohibitive – due to the large size of the servers, something AAM was not setup to cater for
- A Cloud-based deployment was attractive for a number of reasons; not least the low cost due to infrequent usage of the cluster back then (a few hours a week)
- AAM had not directly used a cloud provider for the hosting of their own servers; this was unprecedented in AAM
- Due to AAM's existing enterprise agreements with Microsoft, Azure was the only realistic option because the lead times for setting up such agreements with another cloud provider like Amazon were prohibitive
- But, MDCS was not formally supported on Azure nor integrated in any way (unlike Amazon EC2)…

Overview of AAM's MDCS Cluster in Azure



Key Points:-

- The Head and Worker Node Virtual Machines (VMs) are started when the cluster is required and stopped once no longer needed. This is done by the users using bespoke, in-house built Powershell scripts. The MDCS Windows Service (mdce) is auto-started on each VM and the cluster comes up in a handful of minutes
- Fixed IP addressing used for VMs to ensure cluster comes up cleanly every time
- No data is stored in Azure. Data passes from the Client Node to the Worker Nodes via the Head Node
- Hosted License Manager utilised
- DNS forwarding between Domain Controllers avoids needing to use FQDNs for the Azure VMs

Conclusions

Benefits achieved

- Timely original delivery of the cluster capability – within the 4 week deadline. Faster than could have been achieved on-prem
- Low cost. <£10 per hour in Azure costs to run the cluster
- Self-service. Users can start & stop the cluster themselves using simple scripts

- Perfectly adequate speed/performance of algo execution

Limitations

- Start & Stop scripts rather simplistic and only run interactively with a user. Would be useful to be able to run in silent mode so that the Stop script could be called from within MATLAB upon completion of the algo execution, for example.
- One algo (LPPL) can't run due to each worker process attempting to write to a file on the AAM fileshare but the segregated AD domains won't permit this.
- Unable to combine our two MDCS licenses (of 64 and 16 workers) into one single cluster of 80. A limitation of using the Hosted License Manager over the on-prem FlexNet one.

Aberdeen

Future opportunities

- Enhance StopCluster script so can be execution from within MATLAB upon completion of the algo execution
- Re-deploy the cluster into the fully-AD integrated Azure environment that AAM now has available and so retire the dedicated AD domain up in Azure
- Utilise emerging Azure VM Template from Microsoft which has MDCS pre-installed
- Codify the cluster setup – e.g., using Chef – so that other environments can be created and torn down at will. Such as for MDCS version upgrade testing, for example
- Industrialise some algo executions into Batch method and utilise Azure Batch emerging capability that Microsoft are working on providing

Thank you / Any questions?

Past performance is not a guide to future returns. The value of investments, and the income from them, can go down as well as up and your clients may get back less than the amount invested.

The views expressed in this presentation should not be construed as advice on how to construct a portfolio or whether to buy, retain or sell a particular investment. The information contained in the presentation is for exclusive use by professional customers/eligible counterparties (ECPs) and not the general public. The information is being given only to those persons who have received this document directly from Aberdeen Asset Management (AAM) and must not be acted or relied upon by persons receiving a copy of this document other than directly from AAM. No part of this document may be copied or duplicated in any form or by any means or redistributed without the written consent of AAM.

The information contained herein including any expressions of opinion or forecast have been obtained from or is based upon sources believed by us to be reliable but is not guaranteed as to the accuracy or completeness.

Issued by Aberdeen Asset Managers Limited which is authorised and regulated by the Financial Conduct Authority in the United Kingdom.