FutureJournal

FUTURE STUDIES RESEARCH JOURNAL:
TRENDS AND STRATEGIES

PROFUTURO: PROGRAMA DE ESTUDOS DO FUTURO
Editor Científico: James Terence Coulter Wright
Avaliação: Double Blind Review, pelo SEER/OJS
Revisão: Gramática, normativa e de layout
Recebido em: 21/02/2016 Aprovado em: 10/10/2016

Técnicas de Big Data e Projeção de Risco de Mercado utilizando Dados em Alta Frequência

Alcides Carlos de Araújo
Doutorando em Administração pela Faculdade de Economia, Administração e Contabilidade da Universidade de São Paulo, FEA/USP, Brasil.
alcides.carlos@gmail.com

Alessandra de Ávila Montini
Professora da Área de Métodos Quantitativos e Informática da Faculdade de Economia, Administração e Contabilidade da Universidade de São Paulo, FEA/USP, Brasil.
amontini@usp.br

RESUMO

O mundo passa por um período denominado de Era dos Dados, em que o universo digital poderá ter um tamanho de 44 zetabytes em 2020. Um dos fatores para o crescimento do número de dados são as operações em alta frequência em bolsas de valores, que cresceram significativamente nos últimos anos. Nesse contexto, torna-se difícil mensurar a volatilidade durante o dia devido à quantidade de negociações em tempo real. Nesse caso, devem-se calcular adequadamente as medidas de volatilidade para que o risco realmente seja percebido pelo operador. O objetivo deste artigo é apresentar uma metodologia para obter a volatilidade futura a partir da extração dos dados e do cálculo da volatilidade por meio de técnicas de Big Data. Para atender ao objetivo, foram analisadas todas as ações existentes no banco de dados da BMF&Bovespa. Neste artigo, foram selecionadas as 10 ações mais negociadas no período entre os anos de 2012 e 2014 para apresentação dos resultados. Na primeira fase, desenvolveram-se as funções para tratamento dos dados e estimação das medidas de risco utilizando-se a linguagem de programação Python. Na segunda fase, utilizou-se o Apache Hadoop e o MapReduce (com o Hadoop Streaming) para o cálculo distribuído da estimação do modelo de volatilidade. Para estimar a Volatilidade Percebida, foram utilizadas séries de preços ponderados pelo volume no intervalo de cinco minutos.

FutureJournal
FUTURE STUDIES RESEARCH JOURNAL: TRENDS AND STRATEGIES

PROFUTURO: PROGRAMA DE ESTUDOS DO FUTURO
Editor Científico: James Terence Coulter Wright
Avaliação: Double Blind Review, pelo SEER/OJS
Revisão: Gramática, normativa e de layout
Recebido em: 21/02/2016 Aprovado em: 10/10/2016

Como método de projeção, foi utilizado o modelo HAR-RV, proposto por Corsi. Como resultado, foram desenvolvidas implementações em Python para estimação da Volatilidade Percebida e implementações em Apache Hadoop e MapReduce (com o Hadoop Streaming) para projeção da Volatilidade. Os resultados das estimativas e projeções ocorreram conforme esperado pela literatura.

PALAVRAS-CHAVE: Big Data. Dados em alta frequência. Volatilidade percebida.

Big Data Techniques and Market Risk Forecasting Using High Frequency Data

ABSTRACT

The world is currently living in the Age of Data, the digital universe size is estimated to be around 44 zettabytes in 2020. A factor that contributes to this number of data increasing is high frequency trading in the exchange markets, as these transactions have grown significantly in the last years. In this background, it is difficult to measure the volatility during a day due the vast number of transactions occurring in real time. Thereby, the volatility measure must be estimated carefully so that risks can really be perceived by the operator. This article puts forward a method to estimate future volatility using Big Data techniques for data extraction and volatility estimation. To carry out the objective, the database of all stocks traded at BMF&Bovespa were analyzed. In this article, the 10 stocks most negotiated between the years 2012 and 2014 were chosen for presentation of results. In the first step, steps for data management and risk metrics estimation were developed using Python programming language. In the second step, the Apache Hadoop and the MapReduce (with Hadoop Streaming) was used for distribute

FutureJournal

FUTURE STUDIES RESEARCH JOURNAL:
TRENDS AND STRATEGIES

PROFUTURO: PROGRAMA DE ESTUDOS DO FUTURO
Editor Científico: James Terence Coulter Wright
Avaliação: Double Blind Review, pelo SEER/OJS
Revisão: Gramática, normativa e de layout
Recebido em: 21/02/2016 Aprovado em: 10/10/2016

computing for volatility forecasting model. The Perceived Volatility was estimated using volume weighted average price series with 5 minutes frequency. The projection model HAR-RV, proposed by Corsi, was used for forecasting. As results, a Python program algorithm was constructed for Perceived Volatility estimation and Apache Hadoop and MapReduce (with Hadoop Streaming) implementations for volatility forecasting were developed. The estimates and forecasts results were similar to expected results in the literature review.

KEY-WORDS: Big data. High frequency data. Realized volatility.

1 INTRODUÇÃO

Nos últimos anos, o volume, a variedade e a velocidade necessária para processar dados mudaram de forma significativa. De acordo com White (2012), estima-se que o tamanho do "universo digital" em 2006 era em torno de 0,18 zetabytes, para 2011 estimava-se em torno de 1,8 zetabytes. Segundo Turner, Gantz, Reinsel e Minton (2014), em 2020, estima-se um tamanho de aproximadamente 44 zetabytes.

Esse crescimento é decorrente do resultado de pesquisas e desenvolvimentos na área de computação paralela e distribuída. Conforme Foster e Kesselman (2003), desde o surgimento de arquiteturas distribuídas como as grades computacionais (*grid computing*), empresas e universidades podem ter acesso a plataformas de computação com poder de armazenamento e de processamento comparáveis às de supercomputadores (antes restritos a um pequeno grupo de grandes empresas e universidades).

O acesso a um conjunto de recursos computacionais compartilhados (muitas vezes disponíveis, mas ociosos a maior parte do tempo) e o barateamento do custo desses dispositivos possibilitaram a criação de sistemas computacionais compostos de muitos nós de processamento e com acesso a milhares de gigabytes de dados (*commodity cluster computing*).

Mais recentemente, o surgimento do modelo de computação em nuvem (*cloud computing*) passou a permitir que qualquer desenvolvedor tenha acesso a um número *quasi* infinito de recursos, sem a necessidade de grandes investimentos iniciais. A indústria de TI rapidamente adotou o modelo de computação em nuvem como solução para seus problemas de escalabilidade e provisionamento.

Garantir que um sistema de computação possa utilizar grandes quantidades de recursos computacionais não é um problema trivial. Sistemas nessas plataformas devem ser ao mesmo tempo escaláveis (ou seja, devem conseguir utilizar de forma eficiente a quantidade de recursos disponíveis em um dado momento) e tolerantes a falhas (se um nó do sistema apresentar algum defeito, o sistema deve automaticamente

perceber o problema e adaptar-se, em geral reexecutando parte do processamento em outro nó disponível).

Conforme Dean e Ghemawat (2008), em 2008, o Google apresentou um modelo de programação denominado MapReduce, que é especialmente interessante para a construção de aplicações desse tipo. A utilização desse modelo simplifica a construção de sistemas escaláveis e tolerantes a falhas. Em particular, o arcabouço (*framework*) mais conhecido que implementa esse modelo – o Apache Hadoop – é utilizado pela indústria e por pesquisadores das mais diversas áreas para o processamento de grandes volumes de dados.

Problemas que geram e analisam uma grande quantidade de dados são conhecidos atualmente como problemas de *Big Data*. O baixo custo proporcionado por essas novas plataformas de computação, a facilidade de programação proporcionada por modelos como o MapReduce e a popularização de tais técnicas fazem com que problemas de *Big Data* apareçam nas mais diversas áreas.

Hoje um problema da área financeira é a análise de grandes bancos de dados de negociações na bolsa de valores. Essas negociações, realizadas em tempo real e denominadas negociações em alta frequência (*High Frequency Trading* – HFT), ocorrem considerando *tick by tick*.

Conforme Portnoy (2011), estratégias de HFT utilizam algoritmos complexos, executados em tempo real, para buscar diferentes oportunidades nas bolsas de valores, como antecipar os preços a serem praticados nas negociações que podem ocorrer entre o período de abertura e fechamento de mercado.

Conforme apresenta Zivot (2005), o uso dos dados oriundos de negociações em alta frequência cresceu significativamente nas pesquisas da área financeira. Os principais motivos são o horizonte de decisão dos algoritmos cada vez menor e a maior precisão das estimativas de volatilidades.

No presente artigo, a proposta é utilizar as ferramentas de *Big Data* para tratamento, análise e projeção de dados em alta frequência. Como objetivo secundário, pretende-se comparar essas projeções para verificar o grau de explicação e acurácia da verdadeira volatilidade observada mensuradas em tempo real.

Analisar os mercados de alta frequência com base em tecnologia de *Big Data* é um tema relevante na literatura. Conforme Aldridge (2010), existe uma demanda significativa em busca de informações relacionadas ao mercado de alta frequência, porém pouco foi publicado para auxiliar o entendimento dos investidores.

O tema está internacionalmente em evidência, em pesquisas realizadas pela *International Organization of Securities Commissions* (IOSCO, 2012a; 2012b) em que foram definidos os principais temas que deveriam orientar a atuação do órgão em 2013. Dentre eles estariam a regulação dos negócios de alta frequência, o impacto da tecnologia e a análise dessas infraestruturas de mercado.

Para Seabra (2014), a área de estudos sobre negociações em alta frequência é um tema polêmico no exterior; contudo, ainda é pouco estudado no Brasil. Conforme o autor, apesar dos poucos estudos, espera-se um crescimento do número de trabalhos acadêmicos relacionados ao tema.

Trabalhos como os de Cappa e Pereira (2010) e Wink Junior e Pereira (2011) afirmam que artigos que utilizam bases de dados de alta frequência são raros para ativos brasileiros, pois as bases de dados eram difíceis de ser obtidas. O presente artigo apresenta uma grande contribuição para a utilização das técnicas de *Big Data* para análise dos dados de alta frequência no mercado brasileiro.

2 REVISÃO DA LITERATURA

As técnicas de *Big Data* podem auxiliar a análise de dados em alta frequência. Como o volume de dados a ser processado é alto, a velocidade de processamento dos dados precisa ser muito alta para que os algoritmos tomem decisões rápidas. Os dados a serem analisados são semiestruturados, pois o espaçamento de tempo entre as negociações é irregular e o valor esperado dos dados é primordial para a tomada de decisão na área financeira.

A capacidade de armazenamento com baixo custo possibilitou que dados de todas as negociações fossem armazenados para análises futuras. A velocidade de análise existente agiliza o processo de tomada de decisão.

Mediante esse desenvolvimento computacional, existe a necessidade de as medidas estatísticas serem aprimoradas para apresentar resultados eficientes e consistentes com esse volume de informação.

Conforme demonstram Andersen et al. (2001), Yan e Zivot (2003) e Boudt, Cornelissen e Payseur (2013), existe uma série de desafios na gestão de dados em alta frequência pois o volume é grande e as informações são irregulares. Boudt et al. (2013) citam, por exemplo, os problemas de gerir uma grande quantidade de observações e estimar os parâmetros em séries temporais com espaço de tempo irregular, como os dados *tick by tick*.

Conforme apresentam os autores, essas séries de preços apresentam características que podem provocar estimações inconsistentes caso seja utilizada uma metodologia inadequada. As principais características apresentadas nas séries são: saltos, sincronização e ruídos de microestrutura. O problema da sincronização tende a ocorrer dado que as séries possuem espaçamentos de tempo diferentes para cada ativo estudado. Os saltos e ruídos são provocados por desequilíbrios de curto prazo no mercado.

Nesta revisão da bibliografia, são apresentados os principais conceitos estudados: algoritmos para negócios em alta frequência, técnicas de *Big Data* e medidas de volatilidade e projeção para dados em alta frequência.

2.1 ALGORITMOS DE ALTA FREQUÊNCIA

Conforme Aldridge (2013), nos estudos de alta frequência, é importante diferenciar *High Frequency Trading* (HFT) de *Electronic Trading* e *Algorithmic Trading* (AT).

O *Electronic Trading* refere-se à capacidade de transmissão de ordens eletronicamente, ou seja, sem uso de telefone, carta ou viva-voz. Conforme Aldridge (2013), as atividades de AT e HFT são subgrupos do *Electronic Trading*; percebe-se na literatura uma grande preocupação em diferenciar o AT do HFT.

Aldridge (2013) e Vuorenmaa (2013) conceituam similarmente AT e HFT, classificando-as como negociações sistematizadas e automatizadas.

Desse modo, as decisões e o envio de ordens de negociação ocorrem por meio de programas de computador que analisam os dados e enviam as ordens, funcionando por dias, semanas ou meses. Porém, o sistema tanto pode ser um AT ou HFT.

Um sistema desenvolvido para AT tem o objetivo de minimizar custos de execução, buscando enviar as ordens de negociação quando variáveis como tempo de execução e tamanho do lote sejam otimizadas. Assim, negociações baseadas em AT podem ou não ser executadas imediatamente.

Um sistema desenvolvido para HFT utiliza algoritmos que geram sinais de negociação e otimizam custos de execução. Conforme Vuorenmaa (2013), os HFTs utilizam estratégias de negociação que maximizam a rentabilidade, e o tempo de entrada e saída da operação pode levar minutos, segundos e até milissegundos.

Aldridge (2013) apresenta cinco características associadas ao HFT: a) algoritmos de rápida execução; b) tecnologia de agilidade: geração de sinais, validação de modelos e execução em tempo supersônico; c) frequência de negociação em microssegundos; d) alto volume negociado com lotes de tamanho pequeno; e) as negociações não podem ser realizadas por operadores de mercado, por exemplo, execução de 200 ordens por segundo.

Para Aldridge (2013), a característica que distingue o HFT do AT é a duração da posição (comprada ou vendida). No HFT, a duração máxima é de um dia, não se mantendo a posição para o período subsequente.

No Quadro 1, são apresentados exemplos de pseudoalgoritmos utilizados para negociação. No Quadro 1a, é apresentado um código demonstrado em Sherstov (2004); no Quadro 1b, é visualizado outro código apresentado em Alvim (2009). Esses códigos exibem uma ideia de como os algoritmos de negociação são desenvolvidos.

```
                    while current-time <3 p.m.
                    do state ← updated trader,
                    market stats; action ←
                    COMPUTE-ACTION(state)
if action ≠ VOID
then place/withdraw orders per action
 withdraw all unmatched orders
while market open          #unwind share
position
                    do state ← updated trader,
                    market stats
                    if share-position ≠ 0
then match up to |share-position| shares
of top order in opposite book
```

(a) Código genérico 1 (Sherstov, 2005)

```
                    while marketIsOpen do
                      state ← updateMarket,
updateWallet
                      orders ←
computeStrategy(state)
                      orders ← valid(orders)
                      if orders ≠ void then

                    (matchedOrders,unmatchedOrder
                    s) ← sendOrders(orders)

                    storeOrders(matchedOrders,unma
                    tchedOrders)
  end if
end while
```

(b) Código genérico 2 (Alvim, 2009)

Quadro 1: Exemplos de algoritmos de negociação

Fonte: Sherstov (2005) e Alvim (2009)

2.2 TÉCNICAS DE *BIG DATA*

O *Big Data*, um dos termos mais importantes da área de análise de dados nestes últimos anos, é definido em termos de 5 Vs (volume, velocidade, variedade, veracidade e valor). Em um problema de *Big Data*, deve-se estimar o volume de dados a serem processados, esse processamento deve ser realizado com grande velocidade, pode-se ter uma ampla variedade de tipos de dados (estruturados, semiestruturados e não estruturados), deve existir veracidade nas informações e pretende-se gerar valor com as informações obtidas.

Diante disso, o *Big Data* trata problemas de coletar, gerenciar e processar grandes bases de dados dentro de um período de tempo tolerável. Desse modo, o volume, a velocidade e a variedade com veracidade são as entradas num sistema; e o valor esperado, o resultado (Guo, 2012).

Sistemas computacionais de *Big Data* precisam gerenciar diversas questões relacionadas ao armazenamento e ao processamento de grandes volumes de dados. Conforme Sammer (2012), para a computação esses problemas não são novos; contudo, nos últimos anos o volume e a variedade de dados, além da velocidade necessária para processar, mudaram de forma significativa.

Um dos modelos de programação mais utilizados para o desenvolvimento de aplicações que analisam e manipulam grandes quantidades de dados é o modelo MapReduce, proposto pelo Google. Sua implementação *open-source*, o Apache Hadoop, fornece um arcabouço (*framework*) de execução de aplicações escritas usando o modelo MapReduce, que é escalável e tolerante a falhas.

O Apache Hadoop permite o armazenamento confiável e distribuído de dados utilizando dispositivos de armazenamento baratos (*commodity hardware*). Para tanto, os dados são armazenados de forma distribuída utilizando-se o sistema de arquivos distribuídos denominado HDFS (*Hadoop Distributed File System*), uma implementação *open-source* do Google *File System*, conforme apresentam Ghemawat, Gobioff e Leung (2003).

O HDFS é um sistema de arquivos distribuídos que realiza atualizações e cargas simultâneas; diante disso, torna-se possível armazenar grandes bancos de dados em grandes conglomerados de máquinas (*clusters*). O arquivo de dados é compartilhado em aglomerados computacionais formados por diversos "nós". O modelo MapReduce possibilita o processamento dos dados armazenados nos nós distribuídos com grande eficiência e escalabilidade.

A ideia principal da execução de uma aplicação que segue o modelo MapReduce é a percepção de que "mover dados" é custoso, mas "mover os programas" para perto de onde os dados estão é barato. Portanto, o Apache Hadoop utiliza os mesmos nós utilizados pelo HDFS para executar a aplicação.

O arcabouço é responsável por copiar o programa, que processará os dados para o computador onde ele se encontra, e por gerenciar e armazenar o resultado desse processamento. Além disso, o arcabouço é capaz de detectar falhas em alguma máquina ou em algum *cluster* e as atividades são redirecionadas para os outros computadores. Desse modo, uma análise é totalmente interrompida se todo o *cluster* entrar em colapso.

Um programa de MapReduce é escrito utilizando duas funções: Map e Reduce. Na função Map, todos os elementos são processados por um método e os elementos não afetam uns aos outros, como pode ser visualizado na Figura 1.

$$map(\{1,2,3,4\},\ (\times 2)) \rightarrow \{2,4,6,8\}$$

Figura 1: Exemplo da função Map

Fonte: Elaborado pelos autores

Na função Reduce, todos os elementos da lista são processados juntos, como é observado na Figura 2. Tanto para o Map quanto para o Reduce a entrada é fixa e a saída é enviada para uma nova lista de dados.

$$reduce(\{1,2,3,4\},\ (\times)) \rightarrow \{24\}$$

Figura 2: Exemplo da função Reduce

Fonte: Elaborado pelos autores

Na Figura 3, é apresentada uma arquitetura de como funcionam as funções Map e Reduce.

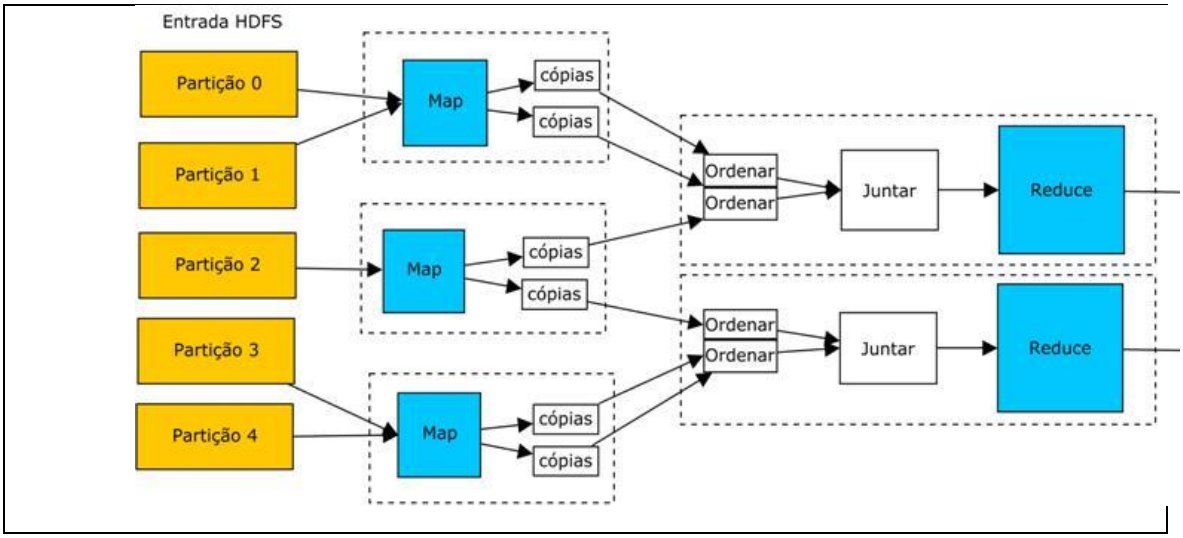


Figura 3: Exemplo da função MapReduce

Fonte: Elaborado pelos autores, com base em White (2012)

2.3 MEDIDAS DE VOLATILIDADE E PROJEÇÃO PARA DADOS EM ALTA FREQUÊNCIA

2.3.1 Medida para estimação da Volatilidade Percebida

Zivot (2005) apresenta uma metodologia de pesquisa para análise de dados de alta frequência, cujo objetivo é estimar, modelar, projetar a volatilidade e projetar a correlação condicional utilizando dados de alta

frequência intradiários. A principal justificativa para o uso de dados em alta frequência ocorre pela maior exatidão das estimativas dos modelos de volatilidade e correlação condicional quando comparados com o uso de dados diários.

Conforme apresenta Aldridge (2010), o processamento dos dados espaçados por tempos regulares, como os dados mensais e diários, é relativamente facilitado devido à existência e à facilidade de aprendizagem dos métodos de estimação. Entretanto, a análise de dados com espaços de tempo irregulares, principalmente com o uso dos dados evento a evento (*tick by tick*), característicos de séries de alta frequência, disponibiliza uma série de oportunidades para análise.

Conforme Andersen et al. (2001), Barndorff-Nielsen e Shephard (2004) e Bauwens, Hafner e Laurent (2012), os estimadores de volatilidade para dados em alta frequência são derivados a partir do conceito da Variância Integrada (*Integrated Variance* – IV).

Seja Y_t um vetor de preços $N \times 1$, em que $Y_t = \ln p_t$, sendo p_t o preço negociado no instante t, a difusão dos preços é expressa na equação 1.

$$\mathrm{d}Y_t = \mu_t \mathrm{d}t + \sigma_t \mathrm{d}W_t, t \geq 0, \quad \textbf{(1)}$$

em que, $\mathrm{d}Y_t$ é o incremento do preço em logaritmo; μ_t é a direção (*drift*) do processo contínuo; σ_t é a volatilidade instantânea de Y_t ;e W_t é um Movimento Browniano Padrão.

Sendo *t* um dia de operações e *i*– *s* o espaçamento entre duas observações, com s = 1 e *i* = 1, …,*M* períodos intradiários, o retorno $r_{t,i}$ é expresso na equação 2.

$$r_{t,i} \equiv Y_{t,i} - Y_{t,i-1} = \int_{t,i-1}^{t,M} \mu_{t,s} \mathrm{d}s + \int_{t,i-1}^{t,M} \sigma_{t,s} \mathrm{d}W_{t,s}, i = 1, \ldots, M. \quad \textbf{(2)}$$

Considera-se que os retornos sejam normalmente distribuídos com distribuição de probabilidade apresentada na expressão 3.

$$r_{t,i} \sim N\left(\mu_{t,M}, IV_{t,M}\right), \quad \textbf{(3)}$$

sendo a média e a variância $IV_{t,M}$ apresentadas nas expressões 4 e 5. A média é dada por:

$$\mu_{t,M} \equiv \int_{t,i-1}^{t,M} \mu_{t,s} \mathrm{d}s\,, \tag{4}$$

a Variância Integrada (*Integrated Variance* – IV) é dada por:

$$IV_{t,M} \equiv \int_{t,i-1}^{t,M} \sigma_{t,s}^2 \mathrm{d}s\,. \tag{5}$$

Para estimar$IV_{t,M}$,, Andersen e Bollerslev (1998) propõem o estimador da Variância Percebida (*Realized Variance* – RV) apresentado na expressão 6.

$$RV_t \equiv \sum_{i=1}^{M} r_{t,i} r'_{t,i}\,. \tag{6}$$

De acordo com Andersen et al. (2003), o estimador foi apresentado com o objetivo de estimar volatilidade utilizando dados de negociação em alta frequência.

2.3.2 Método de projeção para a Volatilidade Percebida

Os modelos preditivos da volatilidade que utilizam dados em alta frequência são amplamente aceitos na academia como os melhores preditores da variância futura, conforme apresentam Boudt et al. (2013). Um dos modelos mais aceitos é o HAR-RV (*Heterogeneous Autorregressive Model of Realized Volatility*).

O modelo HAR-RV foi apresentado em Corsi (2003) como uma adaptação dos modelos *Heterogeneous Autorregressive Conditional Heteroskedasticity* (HARCH) propostos em Müller et al. (1997) e Dacorogna, Müller, Pictet e Olsen (1998). Após a proposta demonstrada em 2003, Corsi (2009) apresenta uma versão revisada.

De acordo com Boudt et al. (2013), o primeiro passo é obter as volatilidades percebidas (RV), por exemplo, por meio da medida apresentada na expressão 6. Durante essa etapa, estima-se o RV para o dia, semana e mês, esses componentes são observados no modelo por representarem os diferentes tipos de participantes do mercado, conforme foi apresentado em Corsi (2003; 2009). O modelo HAR é expresso na equação 7.

$$RV_{t,t+h}^{(d)} = \beta_0 + \beta^{(d)} RV_t + \beta^{(w)} RV_t^{(w)} + \beta^{(m)} RV_t^{(m)} + \epsilon_{t,t+h}. \quad \textbf{(7)}$$

em que, os parâmetros $\beta^{(d)}$, $\beta^{(w)}$ e $\beta^{(m)}$ são os coeficientes para os horizontes de tempo diário, semanal e mensal respectivamente e $RV_t^{(d)}$, $RV_t^{(w)}$ e $RV_t^{(m)}$ são as séries de variâncias com frequência diária, mensal e semanal.

O $\beta^{(d)}$ representa o impacto do investidor de curto prazo; o $\beta^{(w)}$ representa o impacto do investidor de médio prazo e o $\beta^{(m)}$ representa o impacto do investidor de longo prazo. As séries $RV_t^{(w)}$ e $RV_t^{(m)}$ são construídas conforme a expressão 8.

$$RV_t^{(\cdot)} = \frac{1}{h}\left[\sum_{i=1}^{h} RV_{t-i}\right], \quad \textbf{(8)}$$

em que, h possui valores iguais a 5 e 20 para $RV_t^{(w)}$ e $RV_t^{(m)}$ respectivamente.

3 ASPECTOS METODOLÓGICOS

Neste artigo são utilizados dados de negociações *tick by tick* de ações cotadas na Bovespa. Esses dados foram obtidos na BMF&Bovespa, por meio de cuja base de dados podem-se analisar não só séries de tempo evento a evento de todas as ações e de todos os contratos de opções e derivativos negociados (preço, volume, corretora), mas também as séries de ofertas de compra e venda.

Com relação às negociações do mercado à vista entre os anos de 2012 a 2014, tem-se em torno de 80GB de dados. Apesar de o armazenamento ser viável dentro de um único disco, o tratamento desses dados é complicado. As séries devem ser organizadas e, para se obterem as medidas de risco, deve-se tomar cuidado com os dias negociados, horários de abertura e fechamento e feriados.

A implementação foi aplicada em todas as ações existentes no banco de dados. Neste artigo, foram selecionadas as dez ações mais negociadas no período entre os anos de 2012 e 2014 para apresentação dos resultados. As ações foram: Petrobras-PN (PETR4), Vale do Rio Doce-PNA (VALE5), Itaú-Unibanco-PN (ITUB4), Bradesco-PN (BBDC4), Petrobras-ON (PETR3), Banco do Brasil-ON (BBAS3), BMF&Bovespa-ON (BVMF3), Itaú S.A.-PN (ITSA4), Vale do Rio Doce-ON (VALE3) e Gerdau-PN (GGBR4).

A análise dos dados foi composta de duas fases. Na primeira, desenvolveram-se as funções para tratamento dos dados e estimação das medidas de risco utilizando-se a linguagem de programação Python. Na segunda fase, utilizou-se o Apache Hadoop e o MapReduce (com o *Hadoop Streaming*) para o cálculo distribuído da estimação do modelo de volatilidade.

A primeira fase foi subdividida em duas etapas. Na primeira, houve desenvolvimento de funções para o tratamento dos dados, ordenação dos horários, verificação dos dias sem negociação/feriados e correção para horário de verão. Na segunda etapa, foram desenvolvidas as funções para estimação das medidas de volatilidade percebida e projeção.

Para estimar a Variância Percebida (expressão 6), foram utilizadas séries *tick by tick*, séries de preços ponderados pelo volume no intervalo de cinco minutos foram geradas para obtenção dos retornos, por meio dos quais a Variância Percebida diária foi estimada. Como horário normal de negociação, considerou-se o período das 10h às 17h; para o horário de verão, o período das 11h às 18h.

Não foram considerados os preços negociados nos primeiros e nos últimos 15 minutos de pregão, dado que são negociados fora do padrão normal. Também foram desconsiderados os preços negociados no *after-market* (pregão com duração de 30 minutos que se inicia 30 minutos após o

fim do horário normal de negociação). Isto também ocorreu porque os preços são negociados fora do padrão normal nesse período.

A raiz quadrada e o logaritmo da Variância Percebida são consideradas medidas de volatilidade. Para estimar esses valores, existem diversas metodologias. Segundo Andersen, Bollerslev e Diebold (2007) e Val, Pinto e Klotzle (2014), as projeções da raiz quadrada da Variância Percebida (expressão 9) podem ser obtidas por meio do modelo HAR-RV apresentado na expressão 7. Da mesma forma, as projeções do logaritmo da Variância Percebida (expressão 10) podem ser obtidas por meio do modelo HAR-RV apresentado na expressão 7.

$$\left(RV_{t,t+h}^{(d)}\right)^{\frac{1}{2}} = \beta_0 + \beta^{(d)}\left(RV_t^{(d)}\right)^{\frac{1}{2}} + \beta^{(w)}\left(RV_t^{(w)}\right)^{\frac{1}{2}} + \beta^{(m)}\left(RV_t^{(m)}\right)^{\frac{1}{2}} + \epsilon_{t,t+h}, \quad \textbf{(9)}$$

$$\log\left(RV_{t,t+h}^{(d)}\right) = \beta_0 + \beta^{(d)}\log\left(RV_t^{(d)}\right) + \beta^{(w)}\log\left(RV_t^{(w)}\right) + \beta^{(m)}\log\left(RV_t^{(m)}\right) + \epsilon_{t,t+h}. \quad \textbf{(10)}$$

Conforme metodologia proposta por Corsi (2003; 2009), os modelos são estimados por mínimos quadrados ordinários. Para avaliar a qualidade dos modelos de regressão, foram observadas as medidas de R^2 ajustado e raiz do erro médio quadrático, apresentada na expressão 11,

$$RMSE = \sqrt{\frac{1}{n-k}\sum_{i=1}^{n}\left(\widehat{RV}_t - RV_t\right)^2}, \quad \textbf{(11)}$$

em que, k é o número de parâmetros estimados, no caso $k = 4$ para todos os modelos.

4 PROGRAMAÇÃO E ANÁLISE DOS RESULTADOS

Para obtenção das séries de volatilidade diária, mensal e anual, utilizou-se a linguagem de programação Python e suas bibliotecas de análise de dados *Pandas*, para manipulação vetorial de dados *NumPy* e de análises estatísticas *statsmodels*. Uma introdução à análise de dados em

Python é disponibilizada em Sheppard (2012). Trechos do código-fonte utilizado para a análise (sem as otimizações nem tratamentos de erros) são apresentados a seguir.

Na Figura 4, apresenta-se parte do código executado durante a fase de *mapping* da execução, na qual os dados brutos sobre cada transação disponibilizados pela BMF&Bovespa são analisados e seus valores são agrupados por símbolo do instrumento, ano e mês (ou seja, essa é a chave emitida pela função Map).

```
def mapper(arquivo):
      def _agrupador(indice):
            # chave emitida pelo map é (simbolo, ano, mes)
            chave = (indice[0], indice[1].year, indice[1].month)
            return(chave)

      bmf = dados_bmf()
      for chave, tabela in bmf.dados.groupby(_agrupador):
            dados = _df_para_string(tabela)
            print("{}\t{}".format(chave,dados))
```

Figura 4: Código função Map

Fonte: Elaborada pelos autores

Após a fase Map, implementou-se a fase Reduce. Nesta etapa foram obtidas as séries de volatilidades percebidas diárias, semanais e mensais. Na Figura 5, é apresentado o código desenvolvido.

```
    def reducer():
        def _read_mapper_output(file, separator='\t'):
            for line in file:
                yield(line.rstrip().split(separator, maxsplit=1))

        # dados_codificados é a tupla gerada pelo mapper (chave, frame)
        for chave, dados_codificados
            in groupby(_read_mapper_output(sys.stdin), itemgetter(0)):
            simbolo, ano, mes = eval(chave)
            bmf = dados_bmf(dados=pd.concat((_string_para_df(frame[1]).reset_index()
        for frame in dados_codificados)))
            RV = bmf.RVd()
            print(_df_para_string(RV))

    def RVd(dados):
        precos = dados[['preco','qde']]
        media_precos = precos.resample('5min',
                                       how = lambda x:
                                       np.ma.average(x['preco'], weights=x['qde']))

        media_precos['preco'].interpolate(method='nearest', inplace=True)
        p = np.log(media_precos['preco'])
        r = np.square(p - p.shift())

        RV = pd.DataFrame()
        RV['dia'] = r.resample(feriados.bmf, how=lambda x: np.sqrt(x.sum()))
        return(RV)

    def RVx(dados):
        dados['semana'] = pd.rolling_mean(dados['dia'], window=5)
        dados['mes']    = pd.rolling_mean(dados['dia'], window=20)
        return(dados.dropna())

    def combine(RVd):
        bmf = dados_bmf(dados=RVd)
        volatilidade = bmf.dados.groupby(level=0).apply(RVx)
```

Figura 5; Código função Reduce e obtenção das volatilidades percebidas

Fonte: Elaborada pelos autores

Na Tabela 1, são apresentadas as análises descritivas, as séries de volatilidades percebidas das ações PETR3 e PETR4 foram as que demonstraram maior volatilidade anualizada no período de análise. Os resultados são similares aos encontrados em Val et al. (2014) e Santos e Ziegelmann (2014) que estudaram as ações PETR4 e VALE5 e IBOVESPA respectivamente.

Tabela 1: Análise descritiva

Ação	Média	Desvio-Padrão	Assimetria	Curtose	Mínimo	Máximo
PETR4	0,152095	0,496394	12,510982	175,771300	0,001216	7,941847
VALE5	0,066283	0,070532	4,564465	28,225910	0,006764	0,638668
ITUB4	0,088582	0,218494	9,957046	116,117720	0,000077	3,237979
BBDC4	0,091867	0,250815	10,436349	124,475840	0,000056	3,798900
PETR3	0,164838	0,476916	12,199780	170,760670	0,002318	7,805860
BBAS3	0,134591	0,449908	13,057566	190,188530	0,003371	7,422998
BVMF3	0,113764	0,281375	13,043440	192,812870	0,001986	4,799739
ITSA4	0,086745	0,204217	9,813314	109,045180	0,000210	2,864786
VALE3	0,077674	0,073276	4,417971	26,871220	0,005375	0,678795
GGBR4	0,091256	0,058056	1,878575	4,751440	0,006786	0,426215

Fonte: Dados da pesquisa

Após a obtenção das séries de volatilidades percebidas, buscou-se estimar o modelo HAR-RV por meio das expressões 7, 9 e 10. Os coeficientes do modelo HAR-RV ajustado para as projeções da variância percebida (expressão 7) para os 10 ativos são apresentados na Tabela 2.

Os coeficientes do modelo HAR-RV ajustado para as projeções da raiz quadrada da variância percebida (expressão 9) para os 10 ativos são apresentados na Tabela 3.

Os coeficientes do modelo HAR-RV ajustado para as projeções do logaritmo da variância percebida (expressão 10) para os 10 ativos são apresentados na Tabela 4.

Tabela 2: Resultados das regressões RV_t

Ação	β_0	$\beta^{(d)}$	$\beta^{(w)}$	$\beta^{(m)}$	R^2 ajust.	RMSE
PETR4	0,0037	-0,0583	-0,3706**	1,6049***	0,1236	0,4743
VALE5	0,0181**	-0,0623	0,6985***	0,1004	0,1535	0,0625
ITUB4	0,0158	-0,0644	0,1351	0,8264***	0,0988	0,2116
BBDC4	0,0141	-0,0603	0,1201	0,8871	0,1163	0,2408
PETR3	-0,0027	-0,0489	-0,3725**	1,6207***	0,1155	0,4577
BBAS3	0,0170	-0,0243	-0,1716	1,2289***	0,1190	0,4314
BVMF3	0,0166	-0,0310	-0,0038	0,9891***	0,1168	0,2700
ITSA4	0,0166	-0,0352	0,1184	0,7922	0,0960	0,1983
VALE3	0,0183**	-0,0808	0,7075***	0,1478	0,1665	0,0647
GGBR4	0,0205***	0,2077***	0,3418***	0,2327 **	**0,2388**	**0,0509**

*Sig. 10%, **Sig. 5%, ***Sig. 1%.

Fonte: Dados da pesquisa

Tabela 3: Resultados das regressões $(RV_t)^{\frac{1}{2}}$

Ação	β_0	$\beta^{(d)}$	$\beta^{(w)}$	$\beta^{(m)}$	R^2 ajust.	RMSE
PETR4	0,0768***	-0,0026	0,2020*	0,5267***	0,2349	0,1963
VALE5	0,0679***	0,0257	0,5755***	0,0904	0,2023	0,0853
ITUB4	0,0703	-0,0207	0,1924**	0,5166***	0,1833	0,1384
BBDC4	0,0566***	0,0086	0,1525	0,5916***	0,2390	0,1400
PETR3	0,0904	-0,0026	0,2031**	0,5067***	0,2041	0,1945
BBAS3	0,0592***	0,0381	0,1224	0,6167***	**0,2809**	0,1790
BVMF3	0,0698***	0,0261	0,2243**	0,4994***	0,2637	0,1376
ITSA4	0,0706***	0,0274	0,1784*	0,4877***	0,1926	0,1320
VALE3	0,0690**	-0,0090	0,5883***	0,1351	0,1967	0,0850
GGBR4	0,0572***	0,2387***	0,3118***	0,2419**	**0,2804**	**0,0732**

*Sig. 10%, **Sig. 5%, ***Sig. 1%.

Fonte: Dados da pesquisa

Tabela 4: Resultados das regressões $\log(RV_t)$

Ação	β_0	$\beta^{(d)}$	$\beta^{(w)}$	$\beta^{(m)}$	R^2 ajust.	RMSE
PETR4	-0,8013***	0,1729***	0,3943***	0,1455	0,3294	0,6917
VALE5	-0,9527***	0,1185**	0,4972***	0,0936	0,2258	0,6126
ITUB4	-1,0720***	0,1011*	0,2133**	0,3527***	0,1934	0,7100
BBDC4	-0,8135***	0,1503***	0,1707*	0,4293***	0,2656	0,6773
PETR3	-0,8263***	0,1129**	0,3919***	0,1751*	0,2751	0,6607
BBAS3	-0,6473***	0,1859***	0,2524***	0,3507***	**0,3568**	0,6732
BVMF3	-0,6444***	0,1481***	0,3217***	0,3052***	0,3246	0,5814
ITSA4	-0,9841***	0,1253**	0,2513***	0,3149***	0,2323	0,6654
VALE3	-0,8317***	0,0699	0,4994***	0,1620	0,2028	0,5753
GGBR4	-0,5255***	0,2594***	0,2884***	0,2650**	0,2979	**0,4837**

*Sig. 10%, **Sig. 5%, ***Sig. 1%.

Fonte: Dados da pesquisa

Nas Tabelas 2, 3 e 4 são apresentados os valores estimados dos betas, dos coeficientes de determinação ajustado (R^2 ajust.) e RMSE. Os modelos HAR-RV que utilizaram o $\log(RV_t)$ demonstraram melhor ajuste em relação aos modelos que utilizaram o RV_te o $(RV_t)^{\frac{1}{2}}$, pois obtiveram maior coeficiente de determinação ajustado para todas as empresas.

O $\beta^{(d)}$ representa o impacto do investidor de curto prazo, o $\beta^{(w)}$ representa o impacto do investidor de médio prazo e o $\beta^{(m)}$ representa o impacto do investidor de longo prazo.

Nota-se pela Tabela 2 que, em cinco das dez ações, o impacto de médio prazo foi significativo ($\beta^{(w)}$) e, em seis das dez ações, o impacto de longo prazo foi significativo ($\beta^{(m)}$).

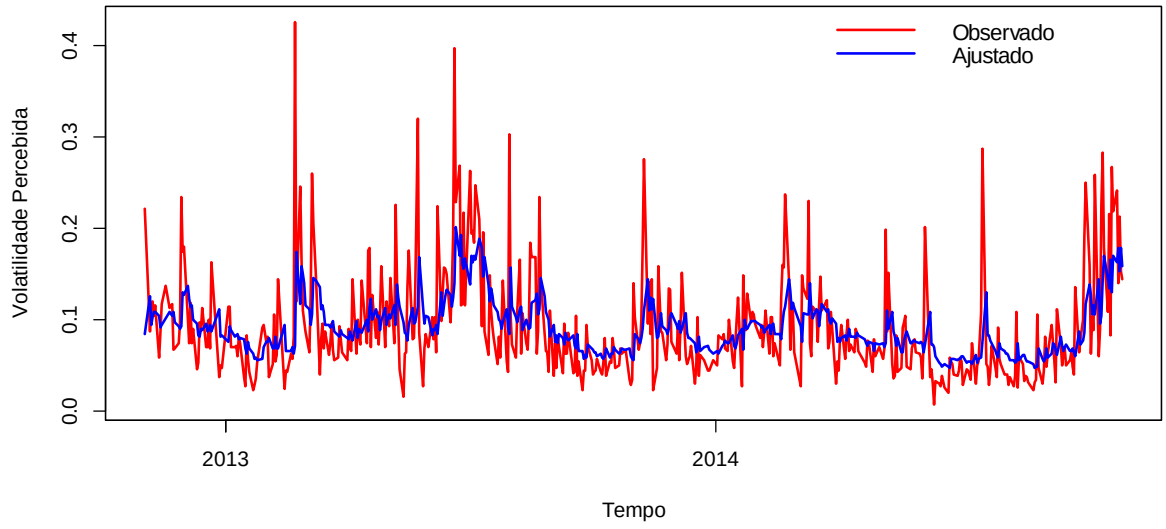
Observa-se pela Tabela 3 que, em oito das dez ações, o impacto de médio prazo foi significativo, assim como o de longo prazo ($\beta^{(m)}$).

Ressalta-se pela Tabela 4 que nove das dez ações apresentaram impacto de curto prazo significante. Todas as ações demonstraram impactos de médio prazo significativos. O impacto de longo prazo foi significativo ($\beta^{(m)}$) em sete das dez ações.

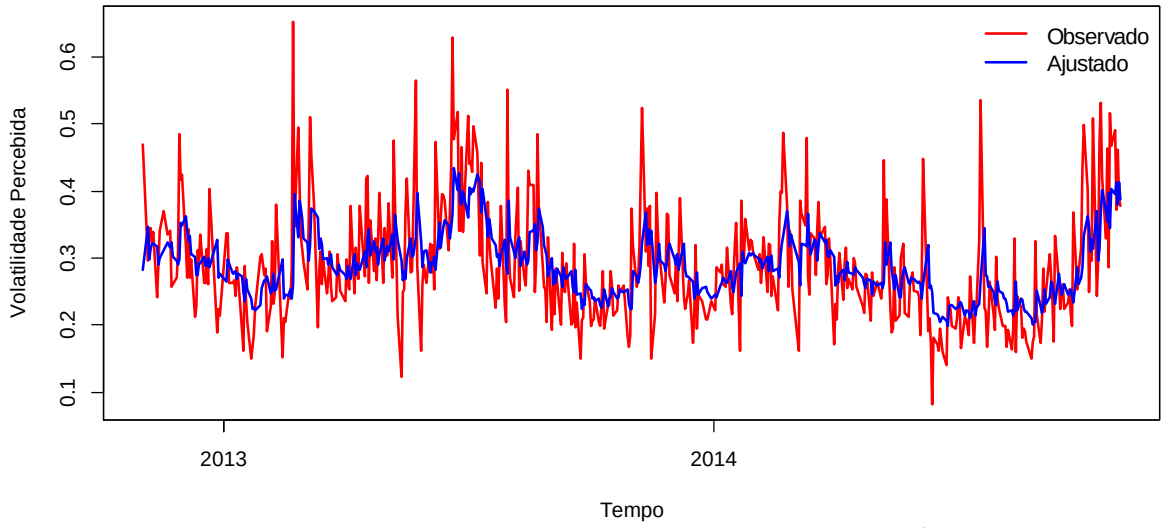
Resultados similares são apresentados nos trabalhos de Andersen et al. (2007) e Val et al. (2014), nos quais os modelos HAR-RV que utilizaram o $\log(RV_t)$ como volatilidade percebida demonstraram melhores ajustes.

Nos Gráficos 1(a), 1(b) e 1(c), são apresentadas as séries originais e ajustadas pelo modelo HAR-RV para as ações que demonstraram os maiores valores de R^2 ajust. para cada modelo ajustado (RV_t, $(RV_t)^{\frac{1}{2}}$, $\log(RV_t)$).

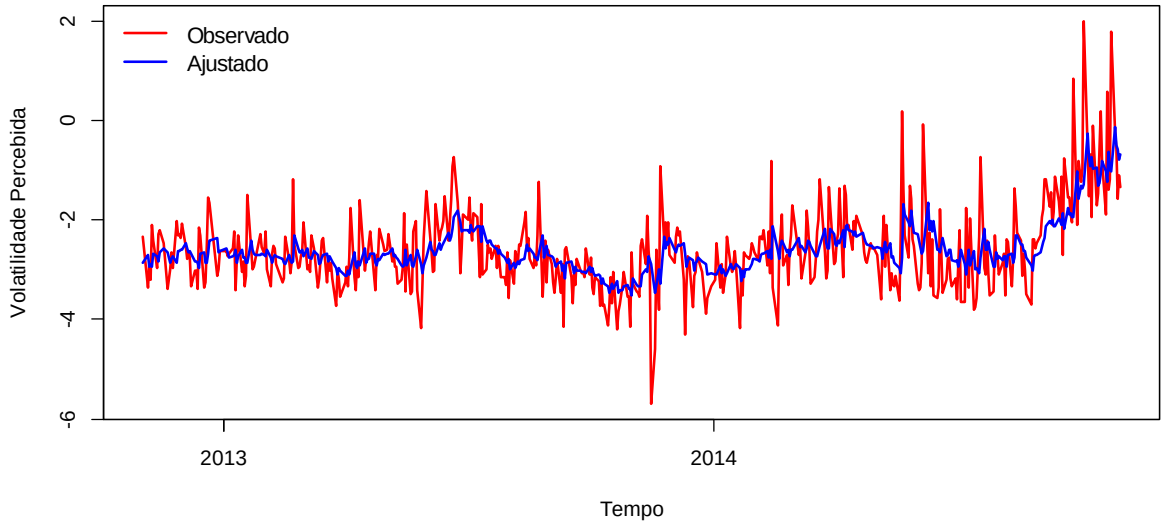
No Gráfico 1(b), foi utilizada a série de GGBR4 dado que o valor de R^2 ajust. foi próximo ao do papel BBAS3, porém o RMSE de GGBR4 foi menor; esse resultado demonstra melhor ajuste para esta série.



(a) Séries original e ajustada GGBR4 para RV_t



(b) Séries original e ajustada GGBR4 para $(RV_t)^{\frac{1}{2}}$



(c) Séries original e ajustada BBDAS3 para $\log(RV_t)$

Gráfico 1: Séries originais e ajustadas

Fonte: Elaborado pelos autores

5 CONSIDERAÇÕES FINAIS

O volume, a variedade e a velocidade necessária para processar dados mudaram de forma significativa, crescimento decorrente do resultado de pesquisas e desenvolvimentos na área de computação paralela e distribuída. Esse desenvolvimento também permitiu que empresas e universidades pudessem ter acesso a plataformas de computação com poder de armazenamento e de processamento comparáveis às de supercomputadores, antes restritos a grandes conglomerados financeiros.

Diante desses novos recursos computacionais, torna-se necessário garantir que um sistema de computação possa utilizar grandes quantidades deles. O arcabouço mais conhecido que implementa esse modelo, denominado Apache Hadoop, é utilizado pela indústria e por pesquisadores das mais diversas áreas para processamento de grandes volumes de dados.

Problemas que necessitam de processamento de grandes volumes de dados são conhecidos atualmente como problemas de *Big Data* e ocorrem nas mais diversas áreas. Hoje, na área financeira, existe o problema de análise de grandes bancos de dados de negociações em bolsa de valores.

Essas negociações, realizadas em tempo real e denominadas negociações em alta frequência (*High Frequency Trading* – HFT), ocorrem considerando *tick by tick*. Conforme Zivot (2005), o uso dos dados oriundos de negociações em alta frequência cresceu significativamente nas pesquisas da área financeira. Alguns dos motivos são o horizonte de decisão dos algoritmos cada vez menor e a melhor precisão das estimativas de risco.

No presente artigo, propôs-se a utilização das ferramentas de *Big Data* para tratamento, análise e projeção de dados em alta frequência, tendo como foco a análise do risco para dados diários mensurados em tempo real.

Como medida de mensuração de risco, foram utilizadas as medidas de Volatilidade Percebida. Para realizar as projeções, foi adotado o modelo HAR-RV proposto por Corsi (2003). Utilizaram-se o Apache Hadoop e o MapReduce (com o Hadoop *Streaming*) para o cálculo distribuído da estimação do modelo de volatilidade.

Como resultados, desenvolveu-se a implementação em MapReduce combinado com a linguagem de programação Python. As medidas de Volatilidade Percebida foram estimadas e o modelo HAR-RV foi ajustado para todas as ações do banco de dados. Neste artigo foram resumidos os resultados das dez ações mais negociadas no período entre 2012 e 2014.

Para futuros trabalhos, recomenda-se ampliar o número de estimadores da Volatilidade Percebida na programação desenvolvida, uma vez que neste artigo foi utilizado somente o estimador da Variância Percebida, proposto por Andersen e Bollerslev (1998) e apresentado na expressão 6. Também se torna relevante ampliar o número de modelos de

projeção da volatilidade, dado que neste artigo utilizou-se somente o modelo HAR-RV.

REFERÊNCIAS

Aldridge, I. (2010). *High-frequency trading: a practical guide to algorithmic strategies and trading systems*. New Jersey: John Wiley & Sons.

Aldridge, I. (2013). *High-frequency trading: a practical guide to algorithmic strategies and trading systems*. New Jersey: John Wiley & Sons.

Alvim, L. G. M. (2009). *Aprendizado de máquina para intraday traders do mercado acionário*. Monografia de Graduação em Ciência da Computação, Pontifícia Universidade Católica do Rio de Janeiro: RJ, Brasil.

Andersen, T. G., & Bollerslev, T. (1998). Answering the skeptics: Yes, standard volatility models do provide accurate forecasts. *International Economic Review*, *34*(4), 885-905.

Andersen, T. G. et al. (2001). The distribution of realized stock return volatility. *Journal of Financial Economics*, *61*(1), 43-76.

Andersen, T. G. et al. (2003). Modeling and forecasting realized volatility. *Econometrica*, *71*(2), 579-625.

Andersen, T. G., Bollerslev, T., & Diebold, F. X. (2007). Roughing it up: including jump components in the measurement, modelling and forecasting of return volatility. *The Review of Economics and Statistics*, *89*(4), 701-720.

Araújo, A. C., & Montini, A. A. (2013). *High frequency trading*: abordagem clássica para análise de preço-volume em uma nova microestrutura de mercado. *Anais do Seminários em Administração - Semead*, 16, São Paulo, SP, Brasil.

Araújo, A. C., & Montini, A. A. (2014). *High frequency trading*: preço, volume e volatilidade em uma nova microestrutura de mercado. *Anais do Seminários em Administração - Semead*, 17, São Paulo, SP, Brasil.

Barndorff-Nielsen, O. E., & Shephard, N. (2004). Power and bipower variation with stochastic volatility and jumps. *Journal of Financial Econometrics*, *2*(1), 1-37.

Bauwens, L., Hafner, C., & Laurent, S. (2012). *Volatility models and their applications*. New Jersey: John Wiley & Sons.

Boudt, K., Cornelissen, J., & Payseur, S. (2013, August). High frequency: toolkit for the analysis of high frequency financial data in R. Recuperado em 19 de agosto, 2014, de http://www2.uaem.mx/r-mirror/web/packages/highfrequency/vignettes/highfrequency.pdf.

Cappa, L., & Pereira, P. L. V. (2010). Modelando a volatilidade dos retornos de Petrobras usando dados de alta frequência. Recuperado em 15 de junho, 2014, de http://bibliotecadigital.fgv.br/dspace/handle/10438/6857.

Corsi, F. (2003). A simple long memory model of realized volatility. Recuperado em 18 de janeiro, 2014, de http://papers.ssrn.com/sol3/papers.cfm?abstract_id=626064. Acesso em 17/03/2014.

Corsi, F. (2009). A simple approximate long-memory model of realized volatility. *Journal of Financial Econometrics*, *7*(2), 174-196.

Dacorogna, M. M., Müller, U. A., Pictet, O. V., & Olsen, R. B. (1998). Modelling short-term volatility with GARCH and HARCH models. In C. Dunis, & B. Zhou, *Nonlinear modelling of high frequency financial time series* (161-176). Chichester, UK: Wiley.

Dean, J., & Ghemawat, S. (2008). MapReduce: simplified data processing on large clusters. *Communications of the ACM*, *51*(1), 107-113.

Foster, I., & Kesselman, C. (2003). *The grid 2: blueprint for a new computing infrastructure*. San Francisco: Elsevier.

Ghemawat, S., Gobioff, H., & Leung, S. (2003). The Google file system. *ACM SIGOPS Operating Systems Review*, *37*(5), 29-43.

Guo, S. (2013). *Hadoop operations and cluster management cookbook*. Birmingham: Packt Publishing.

International Organization of Securities Comissions - IOSCO. (2012a). *Regulatory issues raised by the impact of technological changes on market integrity and efficiency*, final report.. Recuperado em 17 de abril, 2013, de http://www.iosco.org/library/pubdocs/pdf/IOSCOPD361.pdf.

International Organization of Securities Comissions - IOSCO. (2012b). *Technological challenges to effective market surveillance issues and regulatory tools* (Relatório de Consulta). Recuperado em 17 de abril, 2013, de http://www.iosco.org/library/pubdocs/pdf/IOSCOPD361.pdf .

Müller, U. A., Dacorogna, M. M., Davé, R. D., Olsen, R. B., Pictet, O. V., & von Weizsäcker, J. E. (1997). Volatilities of different time resolutions – analyzing the dynamics of market components. *Journal of Empirical Finance*, *4*(2), 213-239.

Pinheiro, M. P., & Gomes, C. F. S. (2008). Evolução do mercado acionário: home broker - Estudo HSBC. *Anais do Simpósio de Excelência em Gestão e Tecnologia - SEGET*, 5, Rio de Janeiro, RJ, Brasil.

Portnoy, K. (2011). High frequency trading and the stock market: a look at the effects of trade volume on stock price changes. *The Park Place Economist*, *19*(1), 35-47.

Sammer, E. (2012). *Haddop operations: a guide for developers and administrators*. Cambridge: O'Reilly.

Santos, Douglas Gomes dos; ZIEGELMANN, F. A. (2014). *Volatility Forecasting via MIDAS, HAR and their Combination: An Empirical Comparative Study for IBOVESPA*. Journal of Forecasting (Print) , v. 33, p. 284-299

Seabra, L. (2014, março 10). Homem x máquina. *Valor.com*. Recuperado em 16 de abril, 2014, de http://www.valor.com.br/financas/3455242/homem-x-maquina#ixzz2vaN82tBT.

Sheppard, K. (2012). *Introduction to Python for econometrics, statistics and data analysis* (version 2). Oxford, Self-published: University of Oxford.

Sherstov, A. A., & Stone, P. (2005). *Three automated stock-trading agents: a comparative study.* In P. Faratin, & J. A. Rodriguez-Aguilar (Eds.), *Agent mediated electronic commerce VI: theories for and engineering of distributed mechanisms and systems* (pp. 173-187). Berlin: Springer Verlag.

Turner, V., Gantz, J. F., Reinsel, D., & Minton, S. (2014). The digital universe of opportunities: rich data and the increasing value of the internet of things. White Paper – IDC iView. Recuperado em 16 de agosto, 2014, de http://www.emc.com/leadership/digital-universe/index.htm.

Val, F. F., Pinto, A. C. F., & Klotzle, M. C. (2014). Volatilidade e previsão de retorno com modelos de alta frequência e GARCH: evidências para o mercado brasileiro. *Revista Contabilidade & Finanças – RC&F*, *25*(65), 189-201.

Vuorenmaa, T. A. (2013). The good, the bad, and the ugly of automated high-frequency trading. *The Journal of Trading*, *8*(1), 58-74.

White, T. (2012). *Hadoop: the definitive guide* (3rd ed.). Cambridge: O'Reilly Media.

Wink Júnior, M. V., & Pereira, P. L. V. (2011). Modeling and forecasting of realized volatility: evidence from Brazil. *Brazilian Review of Econometrics*, *31*(2), 315-337.

Yan, B., & Zivot, E. (2003). *Analysis of high-frequency financial data with s-plus* (Technical Report UWEC-2005-03).

Zivot, E. (2005). *Analysis of high frequency financial data: models, methods and software.* Part II: Modeling and forecasting realized variance measures (Technical Report UWEC-2005-03).