Em nuvens *Infrastructure as a Service* (IaaS), os clientes alugam máquinas virtuais hospedadas em infraestruturas de provedores. O preço do serviço é estabelecido em função da capacidade requisitada dos recursos virtuais. Erros no dimensionamento podem afetar o desempenho das aplicações, ou podem causar desperdício de recursos. Este trabalho investiga o uso de séries temporais para previsão de demanda de recursos de máquinas virtuais. Foi proposta uma metodologia para diagnóstico de modelos de séries temporais sob a perspectiva do provisionamento. Os resultados da metodologia mostraram que os modelos obtidos podem ser usados para provisionar recursos de forma proativa, e que as métricas propostas permitem que um cliente selecione o modelo mais adequado priorizando o desempenho ou o custo de provisionamento.

Orientador: Rafael Rodrigues Obelheiro

Joinville, 2016

ANO
2016

DILSON ASCYNDINO MOREIRA JUNIOR | PREVISÃO DE DEMANDA DE RECURSOS EM NUVENS IAAS USANDO SÉRIES TEMPORAIS

UNIVERSIDADE DO ESTADO DE SANTA CATARINA – UDESC
CENTRO DE CIÊNCIAS TECNOLÓGICAS – CCT
CURSO DE MESTRADO EM COMPUTAÇÃO APLICADA

DISSERTAÇÃO DE MESTRADO

PREVISÃO DE DEMANDA DE RECURSOS EM NUVENS IAAS USANDO SÉRIES TEMPORAIS

DILSON ASCYNDINO MOREIRA JUNIOR

JOINVILLE, 2016

UNIVERSIDADE DO ESTADO DE SANTA CATARINA - UDESC
CENTRO DE CIÊNCIAS TECNOLÓGICAS - CCT
CURSO DE MESTRADO EM COMPUTAÇÃO APLICADA

DILSON ASCYNDINO MOREIRA JUNIOR

PREVISÃO DE DEMANDA DE RECURSOS EM NUVENS IAAS USANDO SÉRIES TEMPORAIS

JOINVILLE

2016

UNIVERSIDADE DO ESTADO DE SANTA CATARINA - UDESC
CENTRO DE CIÊNCIAS TECNOLÓGICAS - CCT
CURSO DE MESTRADO EM COMPUTAÇÃO APLICADA

DILSON ASCYNDINO MOREIRA JUNIOR

PREVISÃO DE DEMANDA DE RECURSOS EM NUVENS IAAS USANDO SÉRIES TEMPORAIS

Dissertação submetida ao Programa de Pós-Graduação em Computação Aplicada do Centro de Ciências Tecnológicas da Universidade do Estado de Santa Catarina, para a obtenção do grau de Mestre em Computação Aplicada.

Orientador:
Prof. Dr. Rafael Rodrigues Obelheiro

JOINVILLE

2016

M838p Moreira Junior, Dilson Ascyndino
Previsão de demanda de recursos em nuvens IaaS usando séries temporais / Dilson Ascyndino Moreira Junior. – 2016.
105 p. : il. ; 21 cm

Orientador: Rafael Rodrigues Obelheiro
Bibliografia: p. 89-92

Dissertação (mestrado) – Universidade do Estado Santa Catarina, Centro de Ciências Tecnológicas, Programa de Pós-Graduação em Computação Aplicada, Joinville, 2016.

1 .Computação em nuvens. 2.Previsão de demanda. 3. Séries temporais. 4. Métricas Obelheiro, Rafael Rodrigues. II. Universidade do Estado Santa Catarina Programa de Pós-Graduação em Computação Aplicada. III. Título.

CDD 004.6782 - 23.ed.

DILSON ASCYNDINO MOREIRA JÚNIOR

PREVISÃO DE DEMANDA DE RECURSOS EM NUVENS IaaS USANDO SÉRIES TEMPORAIS

Dissertação apresentada ao Curso de Mestrado Acadêmico Computação Aplicada como requisito parcial para obtenção do título de Mestre em Computação Aplicada na área de concentração "Ciência da Computação".

Banca Examinadora

Orientador:

Prof. Dr. Rafael Rodrigues Obelheiro
CCT/UDESC

Membros

Prof. Dr. Maurício Aronne Pillon
CCT/UDESC

Prof. Dr. Aldelir Fernando Luiz
IFC-Blumenau/SC

Joinville,SC, 28 de julho de 2016.

Este trabalho é dedicado a Vera, Guilherme e Carolina.

AGRADECIMENTOS

Inicialmente eu gostaria de agradecer ao apoio fundamental dos Promotores de Justiça Dra. Simone Cristina Schultz Corrêa, Dra. Ângela Valença Bordini, Dr. Giovani Werner Tramontin, Dr. Alexandre Piazza, Dr. Sérgio Ricardo Joesting, Dr. Marcelo Mengarda e Dr. Ricardo Paladino.

Aos colegas da UDESC minha gratidão; Marcelo Pereira e Ricardo Pfitscher. Aos professores Dr. Maurício Aronne Pillon, Dr. Guilherme Koslovski pela contribuições e aos servidores da UDESC pelo suporte e a Edson Luiz da Silva pela contribuição essencial.

Gostaria de agradecer imensamente à minha família pela paciência, pelo carinho imenso e pelo apoio recebido para a realização deste trabalho.

Às minhas queridas irmãs Claudia e Fernanda pelo afeto e por estarem sempre presentes.

Aos meus filhos amados Guilherme e Carolina pelo afeto e por compreenderem as ausências.

À minha amada Vera pelo apoio em tudo desde sempre.

Agradeço ao orientador Prof. Dr. Rafael R. Obelheiro pela confiança depositada, disposição em ajudar e pela dedicação no trabalho de orientação.

"Essentially, all models are wrong, but some are useful."

(George E. P. Box)

RESUMO

Em nuvens *Infrastructure as a Service* (IaaS), clientes alugam máquinas virtuais (MVs) hospedadas em infraestruturas computacionais mantidas por provedores de serviço. Tipicamente, o preço do serviço é estabelecido em função da capacidade requisitada dos recursos virtuais (processador, memória, rede e disco), definida pelo próprio cliente, e a tarifação é realizada por hora, com base nos recursos alocados. Isso possibilita que um cliente adapte dinamicamente a capacidade de suas MVs em função da carga, mas o correto dimensionamento dos recursos permanece um desafio devido à natureza das variações de carga. Erros no dimensionamento podem afetar o desempenho das aplicações, no caso de subprovisionamento, ou podem causar desperdício de recursos que permanecerão ociosos, no caso de superprovisionamento. Este trabalho investiga o uso de séries temporais para previsão de demanda de processador e memória em MVs. Foi proposta uma metodologia para obter modelos a partir de dados de monitoração de recursos, e foram criadas métricas para diagnóstico dos modelos obtidos sob a perspectiva do provisionamento. A metodologia proposta foi avaliada usando dados de servidores virtualizados em produção e diferentes métodos de previsão. Os resultados mostraram que os modelos obtidos podem ser usados para provisionar recursos de forma proativa, e que as métricas propostas permitem que um cliente selecione o modelo mais adequado priorizando o desempenho ou o custo de provisionamento.

Palavras-chaves: nuvem computacional, recursos, métricas, monitoramento, séries temporais, previsão.

ABSTRACT

In Infrastructure as a Service (IaaS) clouds, customers rent virtual machines (VMs) hosted in computing infrastructures maintained by cloud platform providers. Usually, service prices are a function of the capacity of virtual resources (CPU, memory, network, and disk) requested by the costumers, and billing is made by the hour, according to the allocated resources. Although this enables customers to change virtual machine capacities to adjust to changing workloads, correctly sizing resources remains a challenge due to the nature of workload variations. Resource sizing errors may affect application performance, in the case of under-provisioning, or lead to resource wastage, in the case of over-provisioning. This work investigates the use of time series for forecasting CPU and memory demand in virtual machines. We propose a methodology for obtaining models from resource monitoring data, and introduce diagnostic metrics for evaluating these models from a provisioning standpoint. The methodology was evaluated using data from virtual servers in production, and different forecasting methods. The results show that the models obtained can be used for proactive resource provisioning, and that the metrics can help a customer to select the most suitable model favoring performance or provisioning cost.

Key-words: cloud computing, computer resources, metrics, monitoring, time series, forecasting.

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE ABREVIATURAS E SIGLAS

ARIMA	Autorregressivo, Integrado, de Médias Móveis (*Autoregressive Integrated Moving Average*)
HW	Holt-Winters
MAE	Erro Absoluto Médio (*Mean Absolute Error*)
NIST	*National Institute of Standards and Technology*
PAR	Previsão Acumulada Relativa
POAP	Percentual de Observações acima da Previsão
SES	Suavização Exponencial Simples (*Simple Exponential Smoothing*)
SLA	Acordo de Nível de Serviço (*Service Level Agreement*)
SLI	Indicadores de Nível de Serviço (*Service Level Indicators*)
SLO	Objetivos de Nível de Serviço (*Service Level Objectives*)
SM	Subestimativa Média

SUMÁRIO

1 INTRODUÇÃO

A computação em nuvem tem sido cada vez mais usada, e as perspectivas apontam para uma manutenção desse crescimento nos próximos anos (CISCO, 2014). Um dos segmentos mais significativos desse mercado utiliza o modelo de IaaS (*Infrastructure as a Service*), no qual os provedores de serviço adquirem e operam infraestruturas computacionais de grande porte e alugam recursos de processamento, rede e armazenamento para seus clientes, sob demanda (MELL; GRANCE, 2011). Tipicamente, um provedor IaaS fornece máquinas virtuais com capacidades (de processamento, rede e armazenamento) determinadas pelo próprio cliente (SULEIMAN, 2012). Logo, para que as infraestruturas virtualizadas na nuvem substituam adequadamente infraestruturas computacionais dedicadas, deve ser contratada a quantidade de recursos exigida pelas aplicações. Como a carga de trabalho tipicamente é variável, a demanda de recursos também é variável.

Para garantir o desempenho desejado, provedores e clientes estabelecem contratualmente acordos de nível de serviço *(Service Level Agreements*, SLAs), que estipulam valores-alvo (chamados objetivos de nível de serviço, ou *Service Level Objectives*, SLOs) para métricas de desempenho dos recursos (chamados indicadores de nível de serviço, ou *Service Level Indicators*, SLIs). Todavia, os SLAs pressupõem que os clientes são capazes de expressar adequadamente as métricas (SLIs) e dimensionar corretamente o nível de serviço (SLO) para cada uma, conforme observado em (PFITSCHER, 2013).

Se os recursos estiverem provisionados abaixo do necessário, compromete-se o desempenho e como consequência ocorrerão violações dos SLOs. Se estiverem provisionados acima do necessário, haverá desperdício de recursos. Portanto, é necessária uma alocação dinâmica de recursos para tratar estes problemas, o que exige a superação de dois desafios. O primeiro desafio é descobrir a quantidade de recursos que

precisa ser alocada, considerando que os requisitos das aplicações variam e que a caracterização do comportamento da aplicação em tempo real é uma tarefa complexa. O segundo desafio é prever esta quantidade para que o ajuste da alocação seja realizado a tempo de atender satisfatoriamente a demanda futura (GONG, 2010).

O planejamento de capacidade pode considerar duas opções simples: a média ou o pico de demanda durante um período de referência. A opção pela média significa um custo menor do serviço, mas implica em problemas de desempenho na ocorrência de picos. A opção pelo pico resolve o problema de desempenho, mas os recursos podem permanecer ociosos durante boa parte do período. Além disso, ao resumir a demanda de um período em um único número, perdem-se informações sobre o comportamento dinâmico da demanda, como tendências de crescimento ou de redução e variações cíclicas, que poderiam ser usadas para estimar com mais precisão os recursos necessários. Consequentemente, o cliente não pode saber se os recursos provisionados para um determinado período estão dimensionados adequadamente, bem como não poderá fazer previsões sobre os níveis exigidos do recurso em períodos futuros. A capacidade de prever a evolução da demanda, além de auxiliar no diagnóstico do provisionamento de recursos, é muito importante em mecanismos de ajuste automático de provisionamento (*auto-scaling*) (LORIDO-BOTRAN, 2014).

As medições periódicas da métrica de um recurso constituem uma série temporal. Uma série temporal pode ser descrita através de um modelo matemático (BOX, 2008; CHATFIELD, 2000; HYNDMAN; ATHANASOPOULOS, 2016), que serve tanto para descrever o comportamento histórico da série quanto para realizar previsões do seu comportamento. Os modelos capturam características da evolução das séries, como tendência e variação periódica, que não podem ser observadas quando se usam estimadores pontuais (tais como média e pico). Portanto, os modelos representam o comportamento da série com maior riqueza de detalhes do que um estimador pontual, o que permite extrair conclusões sobre

os dados com maior grau de certeza.

A ideia geral deste trabalho é elaborar uma metodologia para obter e comparar previsões de demanda utilizando séries temporais com métricas de recursos computacionais para auxiliar o cliente na tomada de decisão: manter, aumentar ou reduzir a quantidade de recurso alocado. O cliente terá o conhecimento, quando houver, das tendências na demanda (crescimento, diminuição ou manutenção), das sazonalidades, de pontos influentes[1] dos períodos nos quais tais comportamentos foram observados e/ou que poderão ocorrer no futuro. Uma vez que há previsões disponíveis, informações sobre tendência e sazonalidade, o cliente poderá ajustar a quantidade de recursos alocados de acordo com os requisitos da aplicação previstos para os diversos períodos. O objetivo do ajuste é a manutenção do desempenho da aplicação em níveis desejados e a redução da ociosidade dos recursos a níveis mínimos. A metodologia para agregação dos dados permite organizar os dados coletados dos recursos de tal forma que as séries temporais obtidas a partir dos dados agregados possam ser processadas em tempos razoáveis sem a perda de informação relevante para efeito de provisionamento.

1.1 Objetivos

O objetivo geral desta dissertação é propor uma metodologia para obter e comparar modelos de previsão de demanda que auxiliem no provisionamento de recursos computacionais virtualizados em nuvens IaaS, considerando a perspectiva de um cliente de nuvem. Esse objetivo geral se desdobra nos seguintes objetivos específicos:

- Elaborar uma metodologia para obter modelos de séries temporais a partir de dados de monitoração de recursos;

[1] Os pontos influentes ou observações atípicas ou *outliers* são anomalias nos dados que fogem ao padrão da distribuição por motivos diversos, tais como, erros de medição, mudança brusca de comportamento, por exemplo (MONTGOMERY; RUNGER, 2011).

- Definir métricas para avaliação de modelos de previsão de demanda de recursos;
- Aplicar a metodologia proposta a dados de servidores virtualizados;
- Comparar os modelos de previsão encontrados com base nos resultados obtidos com os servidores virtualizados.

1.2 Metodologia da Pesquisa

Para (KAUARK, 2010) é importante conhecer os tipos de pesquisa existentes para poder definir os instrumentos e os procedimentos adequados que o pesquisador precisará utilizar no planejamento e na elaboração do trabalho.

Nesta seção este trabalho será classificado de acordo com o raciocínio lógico, a natureza, a abordagem do problema e o objetivo geral.

Segundo (LAKATOS; MARCONI, 2008), existem quatro métodos de pesquisa científica de acordo com o raciocínio lógico: *indutivo*, *dedutivo*, *hipotético-dedutivo* e *dialético*. O método indutivo tem três etapas: observação e análise para descobrir as causas; comparação entre os fatos ou fenômenos para descobrir relações mútuas; e generalização da relação encontrada. As etapas do método hipotético-dedutivo são: formulação das hipóteses; inferência das consequências das hipóteses por dedução; e teste das consequências por experimentação para confirmação ou refutação. O método dialético considera que as coisas estão em constante transformação devido à reciprocidade; existência da negação da negação; que as transformações podem produzir mudança qualitativa. O método desta dissertação é hipotético-dedutivo porque foi formulada a hipótese na qual seria possível fazer previsões de demanda usando séries temporais para o provisionamento de recursos e as consequências da hipótese foram confirmadas após a análise dos dados coletados.

De acordo com (KAUARK, 2010), sob o ponto de vista da natureza, a pesquisa pode ser *básica* ou *aplicada*. A pesquisa é classificada

como básica quando o novo conhecimento obtido não tem uma aplicação prevista e envolve interesses e verdades universais. É aplicada quando o conhecimento obtido tem aplicação prática, na qual se busca a solução de problemas específicos. Esta pesquisa pode ser classificada como pesquisa aplicada, pois seu objetivo é gerar conhecimento (uma metodologia para a previsão de demanda) para a solução de um problema específico (provisionamento de recursos).

Para (KAUARK, 2010), sob o ponto de vista da abordagem, a pesquisa pode ser *qualitativa* ou *quantitativa*. A pesquisa qualitativa não requer o uso de métodos e técnicas estatísticas. É quantitativa quando utiliza recursos e técnicas estatísticas para traduzir e classificar as opiniões e as informações. Esta dissertação é uma pesquisa quantitativa porque utiliza ferramentas estatísticas e modelos matemáticos para fazer a análise das séries temporais.

De acordo com (GIL, 2002), as pesquisas podem ser classificadas de acordo com os objetivos gerais em *exploratórias*, *descritivas* e *explicativas*. Na exploratória o pesquisador busca familiarizar-se com o problema e construir hipóteses. A pesquisa descritiva descreve as características de determinada população, fato ou fenômeno; apresenta respostas sem justificá-las; estabelece padrões ou relações. A pesquisa explicativa busca uma explicação; apresenta as causas e as consequências e explica os mecanismos, os motivos e os processos. Esta pesquisa é do tipo explicativa porque busca uma explicação para o problema da previsão de demanda, as consequências do problema – que são problemas de desempenho e de custo para os clientes –, e como a utilização de séries temporais pode prever o comportamento da demanda.

1.3 Organização do Texto

Esta dissertação está organizada em seis capítulos.

No Capítulo 1 são introduzidos o contexto e o problema da pesquisa, os objetivos do trabalho e a caracterização metodológica.

No Capítulo 2 são apresentados os fundamentos teóricos de nuvem computacional e de previsão usando séries temporais necessários ao entendimento do trabalho.

No Capítulo 3 são discutidos os principais trabalhos relacionados com previsão no ambiente de nuvens computacionais.

No Capítulo 4 é proposta uma metodologia para obtenção de modelos de previsão de demanda, que inclui uma estratégia para agregação de dados e a definição de métricas para avaliação dos modelos.

No Capítulo 5 são apresentados e discutidos os resultados obtidos pela aplicação da metodologia proposta a dados de monitoração de servidores virtualizados em ambiente de produção.

Por fim, no Capítulo 6 são descritas as conclusões obtidas nesta dissertação e apontados trabalhos futuros que podem ser desenvolvidos.

2 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo são apresentados os fundamentos teóricos necessários para a compreensão do problema da pesquisa. A Seção 2.1 discute nuvens computacionais IaaS com foco na visão do cliente. As Seções 2.2 e 2.3 introduzem conceitos de previsão e de séries temporais, respectivamente. Os métodos de previsão utilizados neste trabalho são apresentados nas Seções 2.4 (métodos de suavização exponencial) e 2.5 (método ARIMA).

2.1 Nuvens IaaS

O *National Institute of Standards and Technology* (NIST) define computação em nuvem como

> "um modelo para permitir o acesso ubíquo, conveniente, sob demanda e remoto a um conjunto de recursos computacionais configuráveis (tais como redes, servidores, armazenamento, aplicações e serviços) que podem ser rapidamente provisionados e liberados com um mínimo de esforço de gerenciamento ou interação com o provedor de serviços." (MELL; GRANCE, 2011)

Na prática, isso significa que provedores de serviços adquirem e mantêm infraestruturas computacionais, geralmente de grande porte. Essas infraestruturas contemplam hardware e, em alguns modelos de serviço, também software.

O modelo de serviço considerado neste trabalho é o de infraestrutura como serviço (*Infrastructure as a Service*, IaaS), no qual o serviço oferecido é provisionamento da capacidade de processamento, de armazenamento, de rede, de memória e de outros dispositivos de infraestrutura. Os clientes têm acesso a máquinas virtuais (MVs) nas quais

controlam o sistema operacional, o armazenamento e as aplicações instaladas (PEARCE, 2013). Ao provedor compete gerenciar a infraestrutura física e decidir onde alocar os recursos virtuais solicitados pelos clientes. Exemplos de provedores que oferecem IaaS são Amazon, Google e Rackspace.

Tipicamente, o cliente de uma nuvem IaaS comercial aluga recursos virtuais com uma determinada capacidade, sendo tarifado em função da quantidade de recursos alocados (que pode ser ajustada periodicamente, em geral a cada hora), e não da quantidade de recursos efetivamente usados (SULEIMAN, 2012). Isso significa que o cliente precisa definir a capacidade desejada para suas MVs, sendo de seu interesse alocar o mínimo necessário de recursos para executar suas aplicações com desempenho adequado. Erros no provisionamento dos recursos podem significar desempenho insuficiente, caso os recursos sejam subprovisionados, ou gastos desnecessários, caso os recursos sejam superprovisionados. Muitos clientes de nuvens IaaS acabam superprovisionando seus recursos, devido a fatores, tais como:

- variabilidade da demanda;
- prejuízos decorrentes da perda de usuários e/ou receita causada por desempenho aquém do desejado;
- falta de conhecimento técnico especializado para estimar melhor a capacidade necessária.

Para se obter um provisionamento mais preciso é necessário dispor de dados de monitoramento das máquinas virtuais que permitam conhecer a demanda dos recursos e a carga de trabalho. Em uma nuvem IaaS, um cliente possui apenas a visão interna de suas MVs (ACETO, 2013), sem acesso a dados da infraestrutura física subjacente nem às MVs que compartilham o mesmo hardware. Isso contrasta com a visão do provedor, que observa a demanda total de recursos de cada máquina virtual mas não tem acesso a dados do seu funcionamento interno. Mesmo

quando são coletados no interior de MVs, os dados de monitoramento podem ser enviados para um sistema distribuído de gerenciamento para fins de monitoramento centralizado e tomada de decisão (ACETO, 2013).

2.2 Previsão

Para melhor compreensão do processo de previsão, na seção 2.2.1 são apresentados os fundamentos de previsão, incluindo os conceitos de modelo e de método. Os aspectos estatísticos da previsão também são abordados neste trabalho e estão descritos na Seção 2.2.2.

2.2.1 Fundamentos de Previsão

Previsão é o processo de estimar valores futuros de uma variável por intermédio de um modelo de série temporal. Uma previsão satisfatória precisa captar os padrões e as relações que existem nos dados históricos. A monitoração das variáveis para construir séries históricas pode afetar aquilo que está sendo previsto. Por exemplo: deseja-se fazer previsões da utilização de CPU[1] de uma aplicação específica coletando-se os dados no próprio computador. É preciso, então (i) compreender como a utilização de CPU é afetada pela aplicação, (ii) medir a utilização de CPU ao longo de um período e (iii) utilizar na monitoração uma ferramenta cujo consumo de CPU não seja significativo.

As previsões que usam séries temporais são úteis quando deseja-se fazer previsão de algo que varia no tempo, como temperatura do processador, por exemplo. Se a temperatura do processador é coletada no tempo, obtém-se a série temporal que pode ser usada para previsão, que por sua vez é baseada em valores passados da variável temperatura e de um componente de erro. Então,

$$y_{t+1} = \mathrm{f}(y_t, y_{t-1}, y_{t-2}, \ldots, \mathrm{erro}).$$

[1] Para efeito deste trabalho considera-se que há uma distinção entre CPU e processador. Processador é um chip físico conectado a um soquete de uma placa (placa-mãe ou placa de processadores) e que contém uma ou mais CPU implementadas (GREGG, 2014).

A relação não é exata porque há variáveis que afetam a temperatura que podem não ter sido consideradas. O componente "erro" considera as variações aleatórias e os efeitos das variáveis que não foram consideradas no modelo.

É importante distinguir os conceitos *método* (ou *abordagem*) e *modelo*. O método de previsão é um algoritmo que fornece uma previsão pontual: um valor único que é uma previsão de uma variável para um tempo futuro. O modelo estatístico fornece um processo estocástico de geração de dados que pode ser usado para encontrar uma distribuição de probabilidade para um tempo futuro $T + h$. O modelo estocástico permite: (i) obter uma previsão pontual calculando-se a média ou a mediana da distribuição de probabilidade, (ii) obter intervalos de previsão com um certo nível de confiança e (iii) estabelecer critérios para comparação de modelos (HYNDMAN, 2008).

De acordo com (HYNDMAN; ATHANASOPOULOS, 2016), o processo de previsão requer algumas etapas. Uma das etapas é a *obtenção da informação* a partir de dados estatísticos, como por exemplo os dados coletados das métricas dos recursos. Na etapa de *escolha e ajuste dos modelos* são obtidos os modelos através de um método específico a partir de premissas e dos dados históricos. O ajuste do modelo consiste em encontrar - geralmente utilizando ferramentas computacionais - os parâmetros que tornem o modelo o mais representativo possível dos dados. Os métodos de previsão que fornecem modelos são discutidos nas Seções 2.4 e 2.5.

2.2.2 Visão Estatística da Previsão

O objetivo de uma previsão é estimar o valor futuro de uma dada variável. Como esse valor é desconhecido (de outro modo a previsão não seria necessária), pode-se tratá-lo como uma variável aleatória. Por exemplo, a memória usada em um sistema durante o próximo minuto apenas será conhecida após este minuto ter transcorrido. Entretanto, conhecendo-se a memória atualmente em uso, é possível estimar com

certa precisão uma faixa de valores dentro da qual recairá o consumo de memória no minuto seguinte. Se o consumo atual de memória for usado para prever o consumo de memória num horizonte maior que um minuto, como três ou quatro horas, é razoável supor que a estimativa seja menos precisa. Quanto maior o horizonte de previsão, maior a probabilidade de ocorrer eventos que podem afetar a previsão e portanto, maior a probabilidade do erro de previsão ser maior que o erro dos horizontes mais curtos. Portanto, considera-se que uma previsão será tão incerta quanto maior for o *horizonte* de previsão.

Uma previsão estatística estima o valor intermediário da faixa de valores que a variável aleatória pode assumir. Alguns modelos permitem obter um *intervalo de previsão*, que indica essa faixa com uma dada precisão. Por exemplo, um intervalo de previsão de 95% corresponde a uma faixa de valores que deve incluir o valor real com uma probabilidade de 95%.

Uma previsão sempre parte de um conjunto de observações passadas. Seja $\mathcal{I}$ esse conjunto e y_i a previsão que se pretende obter. Então, $y_i|\mathcal{I}$ é a variável aleatória y_i dado o que se sabe em $\mathcal{I}$. O conjunto dos valores que a variável aleatória pode assumir é a *distribuição de probabilidades* de $y_i|\mathcal{I}$, que, por tratar-se de previsão pode ser denominada de *distribuição de previsão*. O termo *previsão*, na prática, refere-se à média da distribuição da previsão de y_i. Esta média é denotada por $\hat{y}_i$ e corresponde à média dos valores que y_i pode assumir considerando-se tudo que se sabe.

Convém especificar com precisão a informação que está sendo usada na previsão. Por exemplo: $\hat{y}_{t|t-1}$ é a média da previsão de y_t considerando todas observações $(y_1, y_2, ..., y_{t-1})$. Analogamente, $\hat{y}_{T+h|T}$ é a média da previsão y_{T+h} considerando as todas observações anteriores $(y_1, y_2, ..., y_T)$, ou seja, é uma previsão de h passos à frente considerando todas as observações até o tempo T (HYNDMAN; ATHANASOPOULOS, 2016).

2.3 Séries Temporais

Quando uma variável é medida sequencialmente no tempo num intervalo fixo (intervalo de amostragem), os dados obtidos formam uma série temporal. Exemplos: utilização do processador, memória utilizada, temperatura da fonte, etc. Numa série temporal as observações, tipicamente, são dependentes das observações adjacentes. Deseja-se conhecer a natureza da dependência entre estas observações.

A *análise de séries temporais* estuda as técnicas para análise desta dependência, incluindo o desenvolvimento de modelos matemáticos e a utilização destes modelos em diferentes áreas de aplicação (BOX, 2008). Geralmente esses modelos consideram que a série representa um *processo estocástico*, no qual o futuro é determinado apenas parcialmente pelos valores passados, ou seja, existe um componente aleatório na série; e descrevem o seu comportamento não de forma determinística, mas de forma probabilística.

Para (CHATFIELD, 2000) os principais objetivos da análise de séries temporais são: *descrição* dos dados através de medidas estatísticas e métodos gráficos; *modelagem* para encontrar um modelo estatístico apropriado para descrever o processo de geração dos dados; e *previsão* para estimar os valores futuros da série.

Formalmente, uma série temporal pode ser representada por

$$\{y_t\} = \{y_1, y_2, ..., y_n\}$$

onde y_n é n-ésima observação da variável y.

De acordo com (BOX, 2008) e (HYNDMAN, 2008), é possível decompor a variável de uma série em componentes, a saber:

- **Tendência**: Movimento de aumento ou redução na média a longo prazo.
- **Sazonal**: Movimento oscilatório de período fixo e conhecido (horário, diário, semanal, mensal, anual,..) no sentido de aumentar ou

diminuir a intensidade do fenômeno. É um padrão que se repete.

- **Cíclico**: Flutuações que ocorrem em períodos não fixos, às vezes desconhecidos e mais longos comparados com o período sazonal.

- **Irregular ou Erro ou Resíduo**: Movimento oscilatório de curta duração, não previsível, que permanece na série após serem removidas as variações de tendência, sazonais e outros efeitos sistemáticos.

Como exemplo, a Figura 2.1 mostra uma série temporal que representa a utilização de CPU de um servidor num período de 18 dias (*observado*) e os seus componentes *tendência*, *sazonal* e *resíduo*. O valor observado num dado instante é a soma dos valores dos componentes no mesmo instante. É possível constatar que neste exemplo a tendência não é constante. A partir do 5º dia a tendência torna-se crescente e a partir do 14º dia, torna-se decrescente. As variações periódicas diárias também estão presentes neste exemplo com as amplitudes variando de -6% a 8%, uma variação de 14 p.p.[2] Verifica-se uma similaridade entre os padrões da série observada e da série do resíduo, confirmando que o resíduo é o componentes que permanece na série após a remoção dos componentes tendência e sazonal.

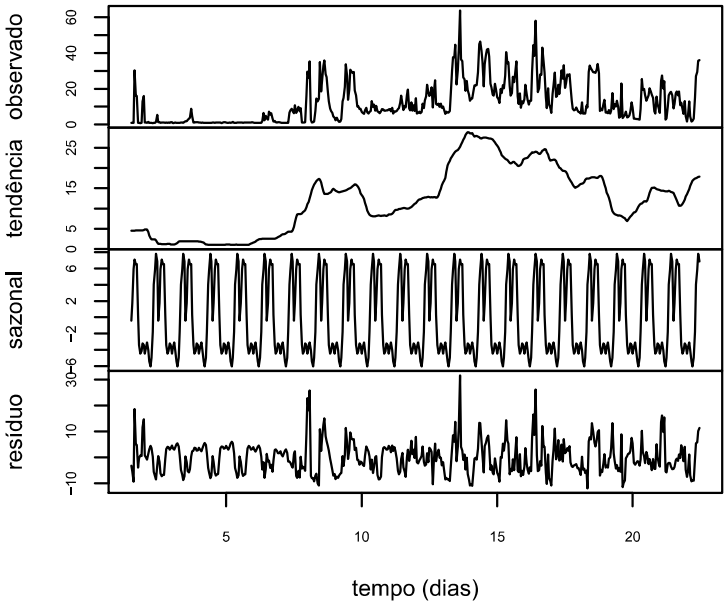
O componente cíclico não será considerado, pois previsões de médio e longo prazo não são objeto de estudo deste trabalho.

2.4 Métodos de Suavização Exponencial

Os métodos de previsão que utilizam suavização (ou alisamento) exponencial usam médias ponderadas de observações passadas, atribuindo o maior peso para as observações mais recentes. Os modelos obtidos caracterizam-se pela facilidade de aplicação, baixo custo de processamento e precisão razoável quando comparados com modelos obtidos

[2] Quando uma variável é medida em percentagens, o valor absoluto entre diferenças da variável é dado em *ponto percentual*, abreviação *p.p.*.

Figura 2.1: Série utilização de CPU em % e seus componentes



Fonte: O autor

por outros métodos; e podem considerar os componentes fundamentais de uma série: tendência, variação sazonal e resíduo (HYNDMAN; ATHANASOPOULOS, 2016).

Nesta seção serão apresentados os métodos suavização exponencial simples (Seção 2.4.1), tendência linear de Holt (Seção 2.4.2), sazonal de Holt-Winters (Seção 2.4.3) e ETS (Seção 2.4.4).

2.4.1 Suavização Exponencial Simples

O método de Suavização Exponencial Simples (*Simple Exponential Smoothing*, SES) pode ser utilizado quando não há tendência ou sazonalidade (efeito do componente variação sazonal definido na Seção 2.3).

A Equação 2.1 é usada para calcular previsões:

$$\hat{y}_{T+1|T} = \alpha y_T + \alpha(1-\alpha)y_{T-1} + \alpha(1-\alpha)^2 y_{T-2} + ..., \quad (2.1)$$

onde $0 \leq \alpha \leq 1$ é o *parâmetro de suavização*. Um exemplo de aplicação deste método é na estimação do *round-trip time* (RTT) de conexões TCP (*Transmission Control Protocol*), que utiliza um modelo SES com $\alpha = 7/8$ (PAXSON, 2011).

A suavização exponencial simples pode ser representada de acordo com seus componentes:

$$\text{Equação de previsão} \quad \hat{y}_{t+1|t} = \ell_t \quad (2.2)$$

$$\text{Equação de suavização} \quad \ell_t = \alpha y_t + (1-\alpha)\ell_{t-1} \quad (2.3)$$

onde ℓ_t é o *nível* (ou valor suavizado) da série no tempo t. A equação de previsão mostra que o valor previsto no tempo $t+1$ é o nível no tempo t. A equação de suavização fornece o nível estimado da série em cada período t. O processo geralmente é inicializado com $\ell_0 = y_1$, onde y_1 é a primeira observação de y.

A equação 2.2 fornece apenas a próxima previsão, chamada *previsão um passo à frente* (*one-step ahead*). Como SES pressupõe que a série não apresenta tendência ou sazonalidade, a previsão para um horizonte $h > 1$ é igual à próxima previsão:

$$\hat{y}_{t+h|t} = \hat{y}_{t+1|t} = \ell_t \qquad h = 2, 3, ... \quad (2.4)$$

2.4.2 Tendência Linear de Holt

O método de *tendência linear de Holt*, também chamado de *suavização exponencial de Holt* (SEH), incorpora ao método SES a tendência como um componente aditivo. São usadas uma equação de previsão (Eq. 2.5) e duas equações de suavização, sendo uma para o nível

(Eq. 2.6) e outra para a tendência (Eq. 2.7):

$$\text{Equação de previsão} \quad \hat{y}_{t+h|t} = \ell_t + hb_t \tag{2.5}$$

$$\text{Equação de nível} \quad \ell_t = \alpha y_t + (1-\alpha)(\ell_{t-1} + b_{t-1}) \tag{2.6}$$

$$\text{Equação de tendência} \quad b_t = \beta^*(\ell_t - \ell_{t-1}) + (1-\beta^*)b_{t-1} \tag{2.7}$$

onde ℓ_t representa a estimativa do nível da série no instante t, b_t a estimativa da tendência (inclinação) da série no instante t, α o parâmetro de suavização para o nível, $0 \leq \alpha \leq 1$, β^* o parâmetro de suavização para a tendência, $0 \leq \beta^* \leq 1$, e h o horizonte de previsão, $h = 1, 2, 3, \ldots$.

O nível ℓ_t é uma média ponderada da observação y_t e da previsão dentro da amostra um passo à frente, dada por $\ell_{t-1} + b_{t-1}$ (Eq. 2.6). A tendência b_t é uma média ponderada da variação da estimativa do nível, dada por $\ell_t - \ell_{t-1}$, e da estimativa anterior da tendência, b_{t-1} (Eq. 2.7).

2.4.3 Holt-Winters

O *método sazonal de Holt-Winters* (HW) suporta sazonalidade. O método tem uma equação de previsão e três equações de suavização, sendo as últimas uma equação para o nível ℓ_t, uma para a tendência b_t e uma para a sazonalidade s_t. O método pode ser Holt-Winters Aditivo e Holt-Winters Multiplicativo, conforme a sazonalidade é aditiva (aproximadamente constante ao longo do tempo) ou multiplicativa (muda de forma proporcional ao nível da série). As equações deste método são omitidas neste trabalho porque são mais complexas que as equações do método linear de Holt, e não são estritamente necessárias para o entendimento do trabalho; elas podem ser encontradas em (HYNDMAN; ATHANASOPOULOS, 2016).

2.4.4 Modelos de Espaço de Estados

Os métodos de suavização exponencial simples, tendêcia linear de Holt e sazonal de Holt-Winters fornecem apenas estimativas pontuais. Em particular, como não possuem um modelo estocástico subjacente,

eles não são capazes de fornecer intervalos de previsão. Visando a preencher esta lacuna, (HYNDMAN, 2002) formularam modelos estatísticos para suavização exponencial. Esses modelos representam um processo estocástico de geração de dados que produz as mesmas estimativas pontuais que os métodos anteriores e ainda descreve a distribuição das previsões. Para tal, os autores elaboraram uma taxionomia.

Os componentes de uma série temporal definidos na Seção 2.3 podem ser expressos como:

- tendência, T_t, sendo $t = 1, 2, ..., n$;
- variação sazonal, S_t, sendo $t = 1, 2, ..., n$;
- erro ou resíduo, ϵ_t, sendo $t = 1, 2, ..., n$;

As possíveis variações na estrutura de um componente do modelo são:

N	Nenhuma estrutura (componente ausente)
A	Estrutura aditiva
A_d	Estrutura amortecida aditiva
M	Estrutura multiplicativa
M_d	Estrutura amortecida multiplicativa

Componentes amortecidos são usados para compensar a propensão de modelos com tendência e/ou sazonalidade de produzir grandes erros de previsão em horizontes longos (HYNDMAN; ATHANASOPOULOS, 2016).

A Tabela 2.1 mostra a taxionomia com as variações de estrutura dos componentes (tendência e sazonal) aplicáveis aos diversos métodos existentes de suavização exponencial, resultando em quinze combinações possíveis (HYNDMAN, 2008).

Cada método conhecido é equivalente a uma das combinações da Tabela 2.1. Exemplo: o método aditivo de Holt-Winters é equivalente à combinação (A,A). Considerando que os erros podem ser aditivos ou multiplicativos, obtém-se um total de trinta combinações.

Tabela 2.1: Variações na Estrutura dos Componentes

Tendência	Sazonal		
	N (nenhuma)	A (aditiva)	M (multiplicativa)
N (nenhuma)	N, N	N, A	N, M
A (aditiva)	A, N	A, A	A, M
A_d (aditiva amortecida)	A_d, N	A_d, A	A_d, M
M (multiplicativa)	M, N	M, A	M, M
M_d (multiplicativa amortecida)	M_d, N	M_d, A	M_d, M

Fonte: Traduzido de (HYNDMAN, 2008).

O modelo de espaço de estados agrega o componente de erro às combinações da Tabela 2.1; este componente é o que permite obter a distribuição das previsões. A notação do modelo utiliza a tupla (E,T,S) para se referir aos componentes erro, tendência e sazonal. Por exemplo, o modelo ETS(A,A,N) possui erro e tendência aditivos, não tem sazonalidade, e corresponde ao modelo de tendência linear de Holt (T=A, S=N) com erros aditivos (E=A).

Cada método de suavização exponencial fornece suas equações pontuais para cálculos recursivos:

- equação de nível, ℓ_t;
- equação de inclinação, b_t;
- equação de sazonalidade, s_{t-m}, onde m é o período sazonal[3];
- equação de previsão para horizonte h, $\hat{y}_{t+h|t}$.

O componente tendência é uma função do nível e da inclinação. O componente sazonal é uma função da sazonalidade, do nível e da inclinação. A equação de previsão é uma função do nível, da inclinação, da sazonalidade e do horizonte de previsão. As equações consideram

[3] O período sazonal denota o número de intervalos de tempo que ocorre a repetição do comportamento. Por exemplo, se uma variável possui sazonalidade diária, um modelo baseado em dados de hora em hora tem $m = 24$, e um modelo baseado em dados de quatro em quatro horas tem $m = 6$.

parâmetros que precisam ser estimados e que variam de acordo com o método de suavização utilizado.

2.5 Método ARIMA

Nesta seção serão apresentados os modelos que compõem o método ARIMA: os autorregressivos, de médias móveis, ARMA e ARIMA. Além dos modelos, também serão apresentados os conceitos de estacionariedade e diferenciação que são necessários para um melhor entendimento do método.

2.5.1 Estacionariedade e Diferenciação

De acordo com (HYNDMAN; ATHANASOPOULOS, 2016; MORETTIN; TOLOI, 2006) uma série é dita estacionária quando suas propriedades não dependem do período de tempo no qual a série é observada. A série se desenvolve no tempo aleatoriamente ao redor de uma média constante. Uma forma de tornar uma série estacionária é tomar sucessivas diferenças da mesma até que se tenha uma série estacionária, processo conhecido como *diferenciação*. Ao tomar as diferenças estabiliza-se a média, eliminando-se a tendência e a sazonalidade.

Uma série diferenciada é a diferença entre observações consecutivas na série original e é expressa por:

$$y'_t = y_t - y_{t-1} \tag{2.8}$$

A diferença sazonal é a diferença entre uma observação e a observação m passos à frente, onde m é o período de sazonalidade, tal que analogamente tem-se:

$$y'_t = y_t - y_{t-m} \tag{2.9}$$

2.5.2 Modelos Autorregressivos

Nos modelos *autorregressivos* (*Auto-Regressive*, AR), a previsão da variável de interesse é realizada através de uma combinação linear de

p valores passados da variável. O modelo é dito autorregressivo de ordem p ou processo AR(p) ou modelo AR(p), e pode ser expresso por:

$$y_t = c + \phi_1 y_{t-1} + \phi_2 y_{t-2} + \cdots + \phi_p y_{t-p} + e_t \tag{2.10}$$

onde c é uma constante, ϕ_i são parâmetros do modelo e e_t são os erros de previsão (componente aleatório com média nula) (HYNDMAN; ATHANASOPOULOS, 2016).

2.5.3 Modelos de Médias Móveis

Nos modelos *de médias móveis* (*Moving Average*, MA), a previsão da variável de interesse é realizada através de uma combinação linear de q erros passados de previsão. O modelo é dito de médias móveis de ordem q ou processo MA(q), e pode ser expresso por:

$$y_t = c + e_t + \theta_1 e_{t-1} + \theta_2 e_{t-2} + \cdots + \theta_q e_{t-q} \tag{2.11}$$

onde c é uma constante, θ_i são parâmetros do modelo e e_t são os erros de previsão (HYNDMAN; ATHANASOPOULOS, 2016).

2.5.4 Modelos ARMA

Combinando-se modelos AR(p) e MA(q) é possível obter modelos com um número menor de parâmetros. A previsão da variável de interesse é realizada através de uma combinação linear de p valores passados da variável e q valores passados do erro de previsão. O modelo é dito autorregressivo e de médias móveis de ordem p, q ou processo ARMA(p,q), e é expresso por:

$$y_t = c + \phi_1 y_{t-1} + \cdots + \phi_p y_{t-p} + e_t + \theta_1 e_{t-1} + \cdots + \theta_q e_{t-q} \tag{2.12}$$

2.5.5 Modelos ARIMA

Os modelos AR(p), MA(q) e ARMA(p,q) são indicados para séries estacionárias. Na prática, a maioria das séries são não estacionárias. Para séries não estacionárias utiliza-se modelos ARIMA (*Auto Regressive*

Integrated Moving Average). Os modelos ARIMA são generalizações dos modelos AR, MA e ARMA e podem ser sazonais ou não sazonais (HYNDMAN; ATHANASOPOULOS, 2016).

O método ARIMA ou método de Box-Jenkins busca as relações que existem entre os dados. O modelo ARIMA não sazonal pode ser expresso por

$$y'_t = c + \phi_1 y'_{t-1} + \cdots + \phi_p y'_{t-p} + \theta_1 e_{t-1} + \cdots + \theta_q e_{t-q} + e_t, \qquad (2.13)$$

onde y'_t é a série diferenciada. Esse modelo é representado pela notação ARIMA(p, d, q), onde d indica quantas vezes a série foi diferenciada, e p e q têm o mesmo significado de um modelo ARMA(p, q). Para obter a série não diferenciada é necessário fazer o processo inverso da diferenciação, chamado de integração, recorrendo à Equação 2.8.

Um modelo ARIMA sazonal pode ter componentes autorregressivos e de média móvel tanto considerando observações imediatamente anteriores quanto considerando observações de períodos sazonais passados. Da mesma forma, a série pode ser diferenciada tanto em relação às observações imediatamente anteriores quanto em relação às observações em períodos sazonais passados. Um modelo ARIMA sazonal é representado por ARIMA$(p, d, q)(P, D, Q)_m$, onde (p, d, q) são os componentes não sazonais, (P, D, Q) são os componentes sazonais, e m é o período sazonal.

2.6 Considerações do Capítulo

Para que clientes de nuvens IaaS obtenham maior benefício na adoção desse tipo de solução, é importante que os recursos virtuais sejam adequadamente provisionados. Um provisionamento inadequado pode se manifestar como insuficiência de capacidade, o que impacta negativamente no desempenho das aplicações hospedadas na nuvem, ou como excesso de capacidade, o que ocasiona custos desnecessários.

Uma possível estratégia para obter um provisionamento adequado

é por intermédio da previsão da demanda de recursos. Como essa demanda raramente é fixa, evoluindo no tempo, ela pode ser interpretada como uma variável aleatória, e modelada usando séries temporais. Modelos de séries temporais fornecem previsões do comportamento das variáveis modeladas em diferentes horizontes de tempo. Existem diversos métodos para obter tais modelos; os modelos usados no presente trabalho foram revisados neste capítulo.

A ideia de aplicar previsão de séries temporais em sistemas computacionais não é propriamente nova. O Capítulo 3 discute a literatura correlata e identifica lacunas não abordadas nos trabalhos relacionados.

3 REVISÃO DA LITERATURA

Modelos baseados em séries temporais são usados de diferentes formas e com diversos propósitos no gerenciamento de recursos em sistemas computacionais. A revisão apresentada neste capítulo concentra-se em trabalhos que aplicam esses modelos para analisar e/ou prever a demanda dos recursos CPU, memória, rede e disco, com ênfase nos dois primeiros em nuvem IaaS.

3.1 Trabalhos Relacionados

Os critérios adotados para classificar os trabalhos estudados foram os seguintes:

- *Objetivo* de uso do modelo. Foram identificados trabalhos que usam modelos para:
  - Previsão de demanda;
  - Diagnóstico de provisionamento;
  - Escalonamento/migração de máquinas virtuais, incluindo a determinação das necessidades de recursos das máquinas virtuais e a avaliação das máquinas físicas atualmente em uso ou candidatas a hospedar máquinas virtuais;
  - Provisionamento automático, quando os recursos alocados são ajustados automaticamente em função da demanda.
- *Recursos* monitorados ou controlados (CPU, memória, rede, etc);
- *Métricas* monitoradas de cada recurso, como utilização de CPU, espaço livre de memória, largura de banda, etc.;
- *Intervalo* de amostragem das métricas consideradas, quando informado. É o intervalo de tempo entre medições sucessivas de uma métrica;

- *Perspectiva* adotada pelo trabalho (cliente ou provedor de nuvem). Para trabalhos no contexto de nuvens computacionais, a perspectiva não apenas determina o principal beneficiário da proposta, mas também afeta quais métricas que podem ser monitoradas. Um cliente pode fazer medições dentro de sua máquina virtual, mas não tem acesso à infraestrutura. Um provedor, por sua vez, pode observar a infraestrutura e a utilização agregada de recursos de máquinas virtuais, mas não pode, de maneira não intrusiva, observar o que acontece dentro de uma máquina virtual de um cliente (ACETO, 2013).

- *Métodos* de previsão avaliados, implementados ou utilizados para realizar o objetivo do trabalho. Mesmo que o objetivo do trabalho não seja exatamente uma previsão, a migração e o provisionamento automático geralmente fazem algum tipo de previsão e, portanto, utilizam um método. Como este trabalho tem foco em previsão usando séries temporais, os métodos foram classificados em ARIMA, suavização exponencial e outros. A categoria ARIMA engloba também modelos AR(p), MA(q) e ARMA(p, q). Do mesmo modo, a categoria de suavização exponencial também abrange os métodos de Suavização Exponencial Simples, Tendência Linear de Holt, Sazonal de Holt-Winters e ETS.

Dentre os trabalhos revisados, a perspectiva de um cliente de nuvem é considerada apenas em (PFITSCHER, 2013; PFITSCHER, 2014). Os demais trabalhos apresentam soluções para tratar de problemas de provedores de nuvem.

Modelos para o diagnóstico de CPU, rede e memória de máquinas virtuais numa nuvem IaaS são introduzidos em (PFITSCHER, 2013; PFITSCHER, 2014). As métricas monitoradas são utilização e *%steal* (tempo de espera da CPU virtual) para CPU, banda consumida e enfileiramento para rede, e memória residente (utilização) e memória reservada (*committed*) para o recurso de memória. O intervalo de amostragem é de um segundo. São fixados limiares para as métricas para determinar se os recursos individuais estão subprovisionados, superprovisionados ou ade-

quadamente provisionados, o que constitui uma ferramenta útil para que um cliente possa tomar decisões sobre as alocações de recursos contratadas junto ao provedor. Os valores das métricas são obtidos dentro da máquina virtual, sem a intervenção do provedor (perspectiva do cliente).

O Sandpiper (WOOD, 2009) faz uso de um algoritmo de detecção de *hotspots*[1] associado a algoritmos para gerência de migração e para provisionamento automático de máquinas virtuais. O algoritmo de detecção constrói perfis de utilização e usa técnicas de previsão baseadas em modelos AR(p) para detectar *hotspots*. Depois determina quando redimensionar ou migrar as máquinas virtuais com *hotspots*. Os recursos monitorados são CPU, rede e memória, e para todos a métrica é utilização, amostrada a cada 10 segundos. A monitoração pode seguir duas abordagens: (i) *black-box*, feita externamente e de forma não intrusiva usando ferramentas de instrumentação e monitoração da plataforma de virtualização; e (ii) *gray-box*, feita internamente, monitorando dentro de cada máquina virtual as estatísticas disponíveis através do sistema operacional em execução na máquina virtual (*guest* ou convidado). O algoritmo de atenuação determina o que migrar, onde migrar e a quantidade de recursos a alocar.

RPPS (FANG, 2012) faz previsão de demanda, provisionamento automático e proativo de recursos e migração de máquinas virtuais. Na arquitetura proposta, um *datacenter* é dividido em grupos com políticas distintas. Os grupos são gerenciados por um controlador de nuvem (*cloud controller*) que contém os módulos *preditor*, *alocador* e *monitor*. O módulo *preditor* faz previsões de demanda usando o método ARIMA. Os recursos considerados são CPU e memória. As métricas são utilização de CPU e de memória, obtidas pelo módulo *monitor* externamente às máquinas virtuais; o intervalo de amostragem não é informado. O provisionamento proativo de máquinas virtuais é realizado de duas formas, em situações

[1] Para (WOOD, 2009) um *hotspot* ocorre em uma máquina física quando a utilização agregada de CPU ou de memória exceder um limiar, ou se a atividade de *swap* ultrapassar um limiar, ou ainda se houver violações do SLA. Para efeito deste trabalho, também é considerado um *hotspot* a variação abrupta da utilização de um recurso numa janela de tempo predeterminada causando sobrecarga.

de demanda normal e na ocorrência de picos repentinos de demanda. Quando a demanda é normal, é feito um ajuste fino dos recursos alocados, segundo a previsão fornecida pelo modelo. Quando ocorrem picos repentinos de demanda, novas máquinas virtuais são adicionadas para atender a situação imprevista. Os resultados experimentais mostraram que as previsões causaram tipicamente menos de 10% de subprovisionamento ou de supervisionamento. Em alguns casos houve superprovisionamento de 20%. Os casos de subprovisionamento que resultam em violações do SLA são contornados com reserva de recursos.

(HUANG, 2012) fazem previsão de demanda para escalonamento de máquinas virtuais na nuvem. Os recursos tratados são CPU e memória, ambos tendo como métrica a utilização. O artigo não menciona o intervalo de amostragem. A previsão utiliza suavização exponencial dupla. O método proposto foi comparado com um método baseada na média e com um modelo de médias móveis ponderadas. Considerando-se que os dados utilizados têm uma tendência linear, não é possivel fazer afirmações seguras sobre a precisão da solução proposta.

O PRESS foi proposto por (GONG, 2010) para fazer previsões de demanda de máquinas virtuais. O recurso avaliado é CPU, a métrica é utilização e o intervalo de amostragem é de um minuto. Para demanda com padrões repetitivos o PRESS obtém uma assinatura do padrão histórico de utilização do recurso e usa esta assinatura para fazer previsões. Para constatar a existência de um padrão repetitivo e fazer previsões deste tipo de demanda, o PRESS usa técnicas de processamento de sinais, a Transformada Rápida de Fourier (*Fast Fourier Transform*, FFT) e o algoritmo *Dynamic Time Warping* (DTW). Para demandas não repetitivas é usada uma cadeia de Markov de tempo discreto com um número finito de estados. Se for detectado um padrão de assinatura na demanda, o PRESS faz as previsões usando a abordagem orientada a assinatura, caso contrário faz as previsões com base na cadeia de Markov.

O Autoflex (MORAIS, 2013) é uma solução de provisionamento automático que utiliza uma técnica híbrida (proativa e reativa). A proposta

é independente de aplicação, pois monitora a utilização dos recursos apenas na infraestrutura na qual as máquias virtuais estão em execução. As diferenças entre os valores medidos pelo módulo *monitor* e os valores de referência são encaminhados para o *controller* que conduzirá a ação adequada para o recurso, como iniciar ou desligar máquinas virtuais. O *controller* tem um comportamento reativo e proativo. O recurso considerado é CPU, e as métricas monitoradas são utilização e número de máquinas virtuais em uso (cada uma tem apenas uma CPU virtual); o intervalo de amostragem é de 5 minutos. Apesar da utilização ser uma métrica de entrada, o modelo de previsão fornece o número de VCPUs necessárias para atender a demanda. Foram utilizados modelos ARIMA e AR, além de outros métodos (regressão linear, autocorrelação e *naïve*[2]). Quando a utilização de um recurso atinge 100%, os autores consideram que há uma violação severa do SLO. Os métodos autocorrelação e ARIMA obtiveram as menores taxas de violações severas de SLO, e os métodos regressão linear e ARIMA os menores custos de provisionamento.

Uma comparação de métodos de previsão sob a perspectiva do provedor foi apresentada em (ENGELBRECHT; GREUNEN, 2015). Os recursos analisados são CPU e memória, ambos tendo como métrica a utilização e com intervalo de amostragem de 5 minutos. Para avaliação dos métodos, os autores definiram as métricas *escore de estimação* e *estimação de sobrecarga*. O escore de estimação considera pesos iguais para as taxas de superestimativas e subestimativas. Uma previsão que tem um escore de estimação de ± 10 é considerada correta. A estimação de sobrecarga classifica as medições como verdadeiro positivo, falso positivo, verdadeiro negativo e falso negativo para obter a taxa de verdadeiro positivo (TPR) e taxa de falso positivo (FPR). Quanto maior a TPR e menor a FPR, melhor a previsão. Os autores concluíram que método AR obteve o melhor desempenho para as métricas propostas, que o método Holt-Winters não obteve um bom desempenho na previsão de sobrecarga

[2] Uma previsão do método *naïve* é sempre igual ao valor da última observação da variável que está sendo prevista, isto é, todos as previsões são y_T, onde y_T é o último valor observado (HYNDMAN; ATHANASOPOULOS, 2016).

de CPU e que o método que utiliza cadeias de Markov obteve o pior desempenho.

A Tabela 3.1 resume os trabalhos relacionados com base nos critérios definidos para esta revisão. As métricas “%cpu”, “%mem” e “%net” representam utilização de CPU, memória e rede, respectivamente, e “n.i.” significa dado não informado.

Tabela 3.1: Sumário dos Trabalhos Relacionados

Trabalho	Grupo					
	Objetivo	Recursos	Métricas	Int.	Visão	Métodos
(PFITSCHER, 2013)	Diag.	CPU Rede	%cpu, %steal (P) banda, fila (R)	1 s	Cliente	-
(PFITSCHER, 2014)	Diag.	Memória	%mem	1 s	Cliente	-
(WOOD, 2009)	Prov. Aut., Migração	CPU Memória Rede	%cpu %mem %net	10 s	Provedor	ARIMA
(FANG, 2012)	Prov. Aut.	CPU	%cpu	n.i.	Provedor	ARIMA
(HUANG, 2012)	Previsão	CPU Memória	%cpu %mem	n.i.	Provedor Cliente	SE
(GONG, 2010)	Previsão	CPU	%cpu	n.i.	Provedor	Outros
(MORAIS, 2013)	Prov. Aut.	CPU	%cpu	5 min	Provedor	ARIMA, outros
(GREUNEN; ENGELBRECHT,2015)	Previsão	CPU Memória	%cpu %mem	5 min	Provedor	AR, SE, outros

Fonte: O autor.

Foram ainda encontrados dois *surveys* (LORIDO-BOTRAN, 2014) e (WEINGÄRTNER, 2014) com intersecção com a revisão desta seção, e que a complementam. O primeiro tem foco em técnicas de provisionamento automático de recursos em nuvem IaaS. As técnicas foram classificadas como *reativas*, quando reagem às mudanças na demanda, ou *proativas*, quando reagem às mudanças e fazem previsões de demanda. As técnicas foram agrupadas em cinco categorias: regras baseadas em limiares estáticos, aprendizagem por reforço, teoria de filas, teoria de controle e análise de séries temporais. A categoria de séries temporais é a única considerada efetivamente preditiva. Para os autores é necessária uma abordagem de provisionamento automático capaz de tratar as variações não previstas na demanda. Eles propõem a utilização de uma

técnica preditiva baseada em algoritmos de previsão com o uso de séries temporais.

Em (WEINGÄRTNER, 2014) é feita uma revisão de modelos e técnicas de previsão e de geração do perfil de utilização de recursos de aplicações (*application profiling*). O trabalho relaciona os principais aspectos computacionais que idealmente deveriam ser suportados por estes tipos de modelos e discute, de forma pouco sistemática, o suporte oferecido por variados trabalhos a tais aspectos. Foi destacada a importância da previsão da utilização dos recursos para o gerenciamento eficiente da nuvem. A natureza dinâmica da nuvem é considerada como o principal obstáculo à criação de modelos para gerenciamento dos recursos.

3.2 Considerações do Capítulo

A análise dos trabalhos relacionados levanta dois aspectos importantes. O primeiro deles é a necessidade de técnicas que façam o ajuste do provisionamento de recursos de forma proativa, ou seja, antecipando mudanças na demanda de recursos. Os trabalhos relacionados consideram principalmente o provisionamento reativo, isto é, após constatação de que houve uma mudança na demanda.

O segundo aspecto relevante é a perspectiva da abordagem. O ajuste no provisionamento é util tanto para o provedor quanto para o cliente da nuvem. Entretanto, constatou-se que a maioria dos trabalhos foram desenvolvidos sob a perspectiva do provedor, enquanto que a perspectiva do cliente é pouco explorada.

O Capítulo 4 introduz uma metodologia para obtenção de modelos de previsão da demanda de recursos em nuvens IaaS. Esses modelos podem ser usados para ajuste proativo do provisionamento de recursos, sob a perspectiva do cliente de nuvem.

4 METODOLOGIA PARA PREVISÃO DE DEMANDA DE RECURSOS

Tendo-se em vista os aspectos pouco explorados nos trabalhos relacionados, nesta dissertação é proposta a previsão da demanda de recursos computacionais sob a perspectiva do cliente da nuvem IaaS. A ideia é que, a partir das previsões de demanda, o cliente possa provisionar os recursos alocados para suas máquinas virtuais de forma proativa.

A previsão de demanda é realizada utilizando análise de séries temporais, que permitem obter modelos da demanda que capturam características como tendência e sazonalidade. Os principais desafios relacionados a essa abordagem são:

1. Como aplicar análise de séries temporais a métricas de demanda de recursos?
2. Dado que existem diversos métodos de análise de séries temporais, como comparar as previsões de demanda obtidas a partir de diferentes métodos?

Este capítulo introduz o procedimento utilizado para obter modelos de previsão da demanda de recursos usando diferentes métodos, e para posteriormente compará-los. O procedimento está dividido em agregação dos dados, descrita na Seção 4.1 e comparação dos métodos de previsão, apresentada na Seção 4.2.

4.1 Agregação dos Dados

Para aplicar um método de análise de séries temporais, é necessário dispor de um conjunto de observações obtidas em intervalos regulares. Portanto, toma-se como premissa que as métricas de demanda de recursos são coletadas utilizando-se um período fixo (chamado *intervalo de amostragem*), e armazenadas em séries históricas.

Tipicamente, intervalos de amostragem situam-se na faixa entre 1 segundo e 15 minutos. Entretanto, intervalos nessa faixa são demasiadamente curtos para análise, devido essencialmente a três fatores:

1. Quanto menor o intervalo de amostragem, maior a variabilidade dos dados, conforme ilustrado na Figura 4.1, que mostra a evolução da ocupação de memória de um servidor amostrada a cada minuto e a mesma métrica agregada em intervalos de 5 e 15 minutos pela média. Isso ocorre porque as métricas na realidade exprimem o valor médio para o intervalo de amostragem, e os efeitos de rajadas de curta duração (máximos/mínimos nos picos/vales, oscilações transientes) são suavizados com o aumento desse intervalo. Séries com alta variabilidade são mais difíceis de modelar do que séries razoavelmente bem comportadas, levando a modelos mais complexos e que ocasionam mais erros de previsão (HYNDMAN; ATHANASOPOULOS, 2016).

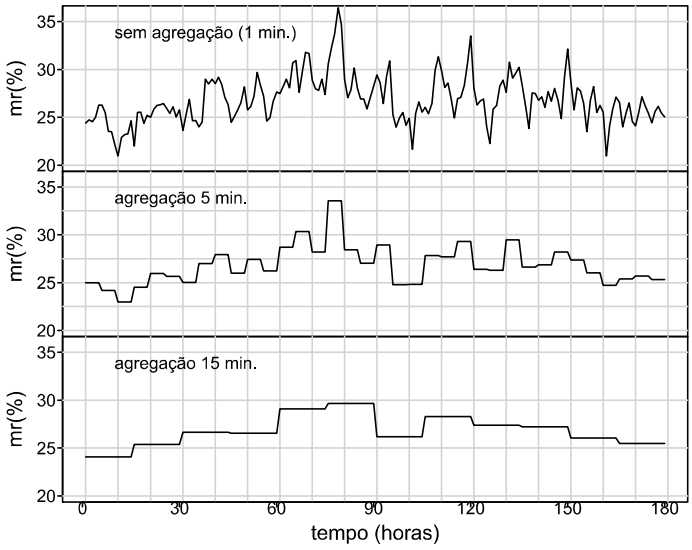
2. Para poder capturar efeitos sazonais é recomendável que a série usada para ajuste do modelo contenha diversos períodos sazonais. Por exemplo, para identificar um comportamento que se repete a cada semana, deve-se ajustar o modelo a partir de algumas semanas de observações. Para um intervalo de amostragem de 5 minutos, tem-se 12 observações por hora, $12 \times 24 = 288$ observações por dia, e $288 \times 7 = 2.016$ observações por semana. Manipular uma quantidade elevada de observações de granularidade fina gera custos de armazenamento e processamento sem necessariamente produzir previsões melhores (MARVASTI, 2011).

3. Além dos fatores relacionados com os dados de entrada, é necessário considerar o propósito do modelo. No contexto de recursos computacionais, a utilidade de se prever a demanda segundo a segundo, ou minuto a minuto, é questionável. Mesmo que seja possível fazer o ajuste da alocação de recursos com esta frequência, provavelmente levará algum tempo até que a nova alocação produza o

efeito pretendido sobre o desempenho (ABDELZAHER, 2008), e é possível que uma nova alocação seja decidida antes que o comportamento do sistema tenha se estabilizado após a alocação anterior.

Considerando-se esses fatores e o fato de que muitos provedores de nuvem permitem modificar a alocação de recursos apenas a cada hora (SULEIMAN, 2012), neste trabalho optou-se por agregar os dados por hora, isto é, resumir a demanda de recursos durante uma hora a partir das métricas observadas originalmente. De acordo com (MARVASTI, 2011), a granularidade de agregação de dados de monitoramento deve ser de até 2,5 horas para evitar uma perda significativa de informação; observa-se que a granularidade adotada é de menos da metade desse limite.

Figura 4.1: Efeito do período de observação sobre a variabilidade das métricas.



Fonte: O autor.

A agregação requer uma função que resuma várias observações

em uma única observação. Geralmente, a função de agregação é a média aritmética; entretanto, provisionar recursos pela média da demanda estimada tende a induzir subprovisionamento – a menos que a métrica tenha uma distribuição assimétrica à direita, com algumas observações discrepantes muito maiores que a maioria dos pontos, que podem ser causadas por um comportamento inesperado do sistema operacional ou de uma aplicação na máquina virtual. Para evitar o subprovisionamento, seria possível usar a maior observação (ou seja, agregar pelo máximo), mas isso tende a causar superprovisionamento, particularmente nos casos em que houvesse observações discrepantes elevadas. Visando chegar a um equilíbrio entre subprovisionamento e superprovisionamento, e considerando que, dentre essas duas situações, o subprovisionamento é mais indesejável, decidiu-se agregar os dados usando um percentil[1]. Foi adotado um percentil de 95%, ou seja, as 5% maiores observações não agregadas serão descartadas para efeito de previsão da demanda, o que implica em 3 minutos num horizonte de 1 hora. O cliente poderá optar por usar um percentil menor (aumentando o percentual de descarte) caso não perceba impacto sobre a aplicação; neste caso, o provisionamento do recurso poderá ser reduzido, o que se traduzirá em economia. No caso oposto, caso o desempenho das aplicações esteja sendo afetado negativamente, o cliente pode aumentar o percentil, reduzindo o percentual de descarte, e consequentemente aumentando o provisionamento do recurso. Para cada hora é esperado ocorrer 5% de subestimativas porque a função de agregação é um percentil 95%. Assim, o percentil é um parâmetro que pode ser ajustado pelo cliente.

4.2 Comparação dos Métodos de Previsão

Para poder comparar o desempenho de diferentes métodos de previsão da demanda de recursos computacionais, adotou-se a seguinte abordagem:

[1] O i-ésimo percentil P_i de um conjunto de observações é o valor tal que i% das observações são iguais ou menores que ele e $(100 - i)$% são maiores (MONTGOMERY; RUNGER, 2011).

1. Obtenção de séries de treinamento e testes para as métricas de interesse;

2. Ajuste de modelos de previsão usando a série de treinamento;

3. Obtenção das métricas de diagnóstico dos modelos;

4. Comparação entre os modelos.

No processo de análise de séries temporais, a série que está sendo analisada é dividida em duas: *série de treinamento* e *série de testes*. A série de treinamento é utilizada para obtenção de modelos e a série de testes para avaliar as previsões obtidas através desses modelos. Para que seja possível capturar efeitos sazonais com período semanal, para cada métrica de interesse deve-se ter uma série de treinamento contendo observações correspondentes a algumas semanas.

O passo seguinte é o ajuste de modelos de previsão usando as séries de treinamento. Os métodos que utilizam suavização exponencial ou ARIMA ou combinações de ambos apresentam previsões satisfatórias, conforme pode ser comprovado pela literatura (MAKRIDAKIS; HIBON, 2000). Por esta razão, foram escolhidos para comparação os métodos Suavização Exponencial Simples (SES), Tendência Linear de Holt, Sazonal de Holt-Winters, ETS e ARIMA, apresentados nas Seções 2.4 e 2.5.

Os resíduos de um modelo ajustado são dados pela diferença entre as observações e os respectivos valores dados pelo modelo:

$$e_i = y_i - \hat{y}_i, \tag{4.1}$$

onde y_i é a i-ésima observação, $\hat{y}_i$ a previsão de y_i e e_i o resíduo (erro) da i-ésima previsão. Como os modelos ajustados geram tanto resíduos positivos (subestimativas) como resíduos negativos (superestimativas), e como subestimativas são indesejáveis pelo impacto negativo que causam no desempenho, ao modelo ajustado é aplicado um fator de correção FC, que é a média dos seus resíduos positivos (que correspondem às

subestimativas). Assim, o modelo de previsão mod_{prev} será dado por

$$e_i^+ = e_i | e_i > 0, \forall i \tag{4.2}$$

$$mod_{prev} = mod_{ajust} + FC = mod_{ajust} + \frac{1}{E^+} \sum_{i=1}^{E^+} e_i^+ \tag{4.3}$$

onde mod_{ajust} é o modelo ajustado (SES, Holt, HW, ETS, ARIMA), E^+ a quantidade de resíduos positivos e e_i^+ os resíduos. Ao final desta etapa, tem-se cinco modelos de previsão diferentes (baseados em SES, Holt, HW, ETS e ARIMA) para cada métrica.

A terceira etapa consiste em obter métricas que permitam fazer um diagnóstico dos modelos ajustados na segunda etapa, o que requer dois passos: (A) obtenção de previsões e (B) cômputo das métricas de diagnóstico dos modelos. Inicialmente, são obtidas as previsões um passo à frente fora da amostra sem reestimação. Uma *previsão um passo à frente* é a previsão da próxima observação (o que corresponde à previsão para a hora seguinte). Como o período da série de testes não faz parte da série de treinamento, tem-se previsões *fora da amostra*. Uma vez que modelo permanece o mesmo ajustado na etapa 2 (ou seja, as observações da série de testes não são usadas para obter um novo modelo), essas previsões são denominadas *sem reestimação* (HYNDMAN; ATHANASOPOULOS, 2016).

O passo (B) do diagnóstico de modelos é a obtenção das métricas de diagnóstico. São usadas quatro métricas:

- erro absoluto médio (MAE, *Mean Absolute Error*);
- percentual de observações acima da previsão (POAP);
- subestimativa média (SM);
- previsão acumulada relativa (PAR).

A primeira é uma métrica clássica de diagnóstico de previsões de séries temporais, e as demais são métricas introduzidas neste trabalho.

O MAE de um conjunto de N previsões é dado pela média do valor absoluto dos resíduos e_i:

$$\text{MAE} = \frac{1}{N}\sum_{i=1}^{N}|e_i|, \tag{4.4}$$

onde y_i é a i-ésima observação, $\hat{y}_i$ a previsão de y_i e e_i o resíduo (erro) da i-ésima previsão. O MAE é uma métrica simples e de fácil compreensão. O melhor método é aquele cujo conjunto de previsões produz o menor MAE. Quando não há observações de valor nulo ou negativo e quando as previsões que serão comparadas estão numa mesma escala, o MAE pode ser usado (HYNDMAN; KOEHLER, 2006).

Assim como outras métricas estatísticas de diagnóstico de previsões, o MAE atribui o mesmo peso para subestimativas e superestimativas. Pelo MAE, um modelo que constantemente subestime a demanda pode ser considerado melhor do que outro que forneça apenas superestimativas – ou superestimativas e subestimativas –, dependendo da magnitude das diferenças observadas. Na previsão de demanda de recursos com fins de provisionamento, porém, considera-se pior subestimar a demanda (o que prejudica o desempenho) do que superestimá-la (o que causa desperdício de recursos). Assim, foi definida uma segunda métrica de diagnóstico, que é o *percentual de observações não agregadas acima da previsão* (POAP). Para obter essa métrica são contadas as observações não agregadas num período de uma hora (período da série de testes) que são maiores que a previsão para aquela hora, repetindo-se o procedimento para todas as horas da série de testes. Ao final soma-se as contagens de todas as horas e divide-se o resultado da soma pelo número total de observações não agregadas da série de testes, multiplicando o resultado por 100 para obter o valor percentual. O melhor método de previsão será aquele que apresentar o menor percentual de observações não agregadas acima da previsão (menor POAP). A métrica POAP_k

de um modelo k é dada por:

$$\text{POAP}_k = 100 \cdot \frac{1}{60N_T} \sum_{i=1}^{N_T} \sum_{j=1}^{60} c_{ij} \tag{4.5}$$

onde j é o j-ésimo minuto da i-ésima hora e c_{ij} é definido como

$$c_{ij} = \begin{cases} 1, & \text{se } x_{ij} > \hat{y}_i \\ 0, & \text{caso contrário} \end{cases} \tag{4.6}$$

onde x_{ij} é a observação não agregada do j-ésimo minuto da i-ésima hora.

Para ilustrar a métrica POAP, na Figura 4.2 são mostradas as observações e as previsões para seis períodos de uma hora para os métodos ETS e ARIMA. É possível constatar visualmente que neste período o percentual de observações acima da previsão foi menor quando a previsão foi realizada pelo método ARIMA; numericamente, $\text{POAP}_{\text{ARIMA}} = 6{,}7\% > \text{POAP}_{\text{ETS}} = 6{,}1\%$. Para este caso, o método de ETS é o mais bem sucedido em evitar o subprovisionamento.

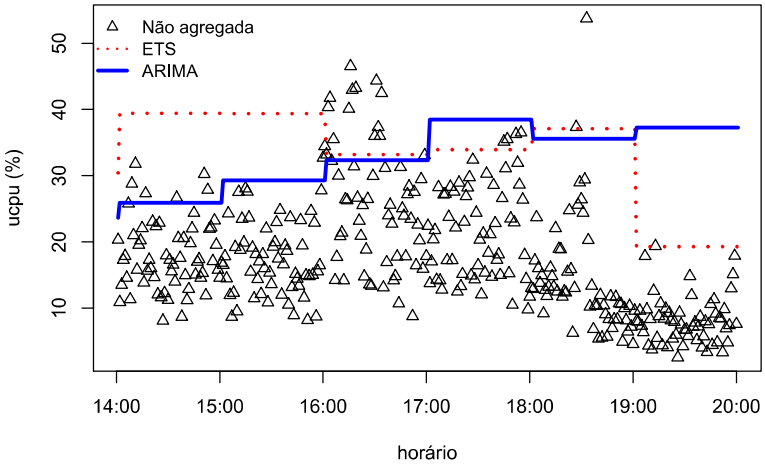
O POAP permite saber a frequência de subestimativas, mas não informa a sua magnitude. Para quantificar a magnitude, foi definida a métrica *subestimativa média* (SM), que representa a média das diferenças absolutas entre o valor previsto e as observações não agregadas acima dessa previsão:

$$\text{S}_{ij} = \max\left(x_{ij} - \hat{y}_i, 0\right) \tag{4.7}$$

$$\text{SM} = \frac{\sum_{i=1}^{N_T} \sum_{j=1}^{60} \text{S}_{ij}}{\#\{\text{S}_{ij} \mid \text{S}_{ij} > 0\}} \tag{4.8}$$

Na Equação 4.8, o numerador representa o somatório das subestimativas encontradas na série de testes, e o denominador representa o número de ocorrências de subestimativas na série. Como as diferenças são absolutas, a subestimativa média é expressa na mesma unidade das observações. Além disso, a exemplo de MAE e POAP, o melhor método será aquele que apresentar a menor subestimativa média.

Figura 4.2: Observações não agregadas e previsões para métodos ETS e ARIMA.



Fonte: O autor.

As métricas POAP e SM quantificam as subestimativas de cada modelo. Minimizar POAP e SM favorece o desempenho das máquinas virtuais, mas esse objetivo deve ser equilibrado com o custo de provisionamento com base nas previsões encontradas. Para quantificar esse custo, foi definida uma quarta métrica de diagnóstico, que é a *previsão acumulada relativa* (PAR). A previsão acumulada PA_k de um modelo k é o somatório das previsões fornecidas pelo modelo ao longo da série de testes (N_T é o comprimento da série de testes):

$$\mathrm{PA}_k = \sum_{i=1}^{N_T} \hat{y}_i \tag{4.9}$$

A previsão acumulada relativa PAR_k do modelo k representa a diferença

percentual entre PA_k e a menor previsão acumulada $PA_{min} = \min(PA_k), \forall k$:

$$PAR_k = 100 \frac{PA_k - PA_{min}}{PA_{min}} \tag{4.10}$$

Para o modelo de menor previsão acumulada, $PA_k = PA_{min}$, e portanto $PAR_k = 0$. A métrica PAR oferece uma aproximação da diferença relativa no custo de provisionamento de diferentes modelos, permitindo assim compará-los. A premissa embutida nessa métrica é que o custo de provisionamento é proporcional a PAR.

Finalmente, na última etapa os modelos são comparados através da análise das métricas obtidas na etapa 3. Essa comparação deve ser feita com cuidado, uma vez que (i) as métricas não têm a mesma relevância e (ii) a magnitude das diferenças entre as métricas é relevante. Por exemplo, se o método A tem $MAE_A = 5$ e $POAP_A = 50\%$ e o método B tem $MAE_B = 4{,}8$ e $POAP_B = 60\%$, A é considerado melhor que B, mesmo com $MAE_A > MAE_B$. Além disso, a avaliação das subestimativas deve considerar POAP e SM em conjunto: um método C que gere subestimativas mais frequentes que um outro método D (ou seja, $POAP_C > POAP_D$) pode ser melhor se a magnitude das subestimativas for menor ($SM_C < SM_D$) – a conclusão vai depender das diferenças entre as métricas.

4.3 Considerações do Capítulo

Neste capítulo foi apresentada uma metodologia para fazer previsão de demanda de recursos computacionais. A metodologia é dividida em duas etapas: agregação e comparação. Na agregação os dados são agregados em períodos de 1 hora pelo percentil 95% . As previsões são obtidas utilizando-se cinco métodos de previsão de séries temporais. As previsões fornecidas por cada método são comparadas utilizando-se as métricas de diagnóstico MAE, POAP, SM e PAR, que oferecem informações sobre o desempenho dos métodos que auxiliam a compreender a qualidade dessas previsões.

O procedimento proposto neste capítulo foi aplicado a dados de servidores virtualizados em ambiente de produção. Essa avaliação experimental e seus resultados são discutidos no Capítulo 5.

5 AVALIAÇÃO

No Capítulo 4 foi proposto um procedimento para obter e comparar modelos de previsão da demanda de recursos usando diferentes métodos. O presente capítulo descreve a avaliação do procedimento proposto com base em conjuntos de dados reais que foram obtidos de servidores em ambiente de produção.

A Seção 5.1 descreve os conjuntos de dados e como esses dados foram tratados para obter séries temporais nos moldes esperados pelo procedimento. A Seção 5.2 explica como foram ajustados os modelos. A Seção 5.3 apresenta e discute os resultados obtidos.

As análises descritas neste capítulo foram realizadas usando o ambiente para computação estatística R (R FOUNDATION, 2016), versão 3.3.1, com o pacote *forecast* (HYNDMAN; KHANDAKAR, 2008), versão 7.1.

5.1 Descrição e Tratamento dos Dados

Foi necessário realizar um tratamento nos dados para a obtenção das séries temporais nas condições propostas no Capítulo 4. O tratamento foi realizado em duas etapas: imputação e agregação.

5.1.1 Descrição

Os dados analisados neste trabalho foram coletados em servidores virtualizados em ambientes empresariais, sendo sistemas em ambiente de produção. Esses dados foram gentilmente cedidos por colaboradores que solicitaram anonimato. Os servidores são identificados[1] por ZAB1, ZMU1, ZMU2, ZMU3 e ZMU4 conforme Tabela 5.1 que resume as

[1] Os nomes dos servidores mencionados não têm qualquer relação com nomes de empresas ou marcas de produtos.

suas principais informações de configuração. A plataforma de virtualização utilizada é o VMware vSphere (VMWARE, 2016), e todos os servidores usam variantes Linux como sistema operacional convidado.

Tabela 5.1: Informações de configuração dos servidores analisados

	Servidor				
	ZAB1	ZMU1	ZMU2	ZMU3	ZMU4
Número de VCPUs	4	4	1	4	4
Memória	4GB	12GB	3GB	16GB	16GB
Arquitetura	x86-32	x86-64	x86-64	x86-64	x86-64
SO Convidado	Linux CentOS v6	Red Hat Enterprise Linux v6.6	Oracle Linux Server v6.4	Linux CentOS v6.4	Linux CentOS v6.4
Principais Serviços	SGBD Firebird	Aplicações Web	Portal Web	SGBD Oracle	WebSense Proxy

Fonte: O autor.

Os dados brutos são métricas dos recursos CPU e memória coletadas com o monitor `sar` (GODARD, 2016). O intervalo de amostragem é 1 minuto, e cada servidor foi monitorado por três semanas, no mínimo, respeitando a duração necessária para permitir a identificação de sazonalidade semanal (conforme discutido na Seção 4.1). A partir dos dados brutos foram obtidas as métricas de provisionamento sugeridas por (PFITSCHER, 2013) e por (PFITSCHER, 2014) para diagnóstico de recursos na nuvem IaaS, a saber: utilização (`ucpu`) e `steal` para CPU, memória residente (`mr`) e memória reservada (`mc`). A partir das previsões destas métricas, o cliente pode alocar a quantidade de recursos para obter um provisionamento adequado, como será mostrado pelos exemplos a seguir.

Inicialmente, seja uma máquina virtual com 800 MB de memória (*MemTotal* = 800), cujas estimativas de `mr` e `mc` são 75% e 140%, respectivamente. De acordo com (PFITSCHER, 2014), a memória está adequadamente provisionada quando 50% $\leq$ `mr` $<$ 70% e `mc` $<$ 150%. Portanto, como a estimativa de `mr` é superior a 70%, o cliente deve aumentar sua alocação de memória para evitar o subprovisionamento. A memória residente estimada é de $800 \times 0{,}75 = 600$ MB. Para que essa quantidade seja

inferior a 70% da memória total (provisionamento adequado), deve-se ter:

$$\frac{MemRes_{t+1}}{MemTotal_{t+1}} < 0{,}7$$
$$MemTotal_{t+1} \geq \frac{600}{0{,}7} = 857{,}1 \approx 858 \text{ MB}$$

Logo, a memória total no tempo $t + 1$, $MemTotal_{t+1}$, para que o provisionamento seja adequado é 858 MB, aproximadamente. Nos cálculos foi desconsiderado o valor de `mc` porque o modelo de diagnóstico de (PFITSCHER, 2014) entende que, neste caso, `mc` é irrelevante.

Agora, seja um exemplo no qual `mc` é relevante e as estimativas de `mr` e `mc` são 80% e 160%, respectivamente, para *MemTotal* = 800 MB. Nesse caso, *MemRes* = $0{,}8 \times 800$ = 640 MB e *MemCommit* = $1{,}6 \times 800$ = 1280 MB. Considerando os limiares para provisionamento de memória, obtém-se:

`mr`: $MemTotal_{t+1} \geq 640/0{,}7 = 914{,}4$ MB

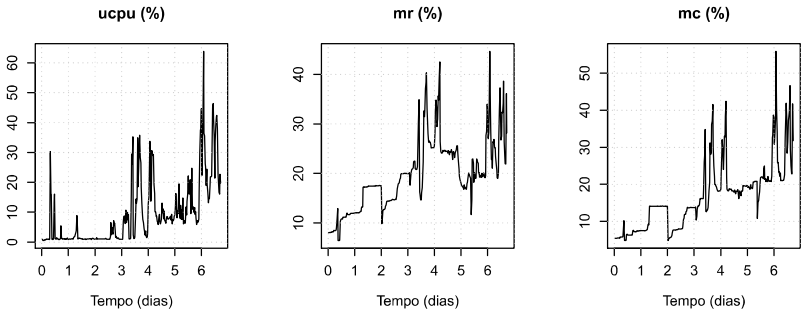
`mc`: $MemTotal_{t+1} \geq 1280/1{,}5 = 853{,}3$ MB

Logo, a memória total no tempo $t + 1$, $MemTotal_{t+1}$, para que o provisionamento seja adequado é aquela que atende as restrições das métricas de provisionamento:

$$MemTotal_{t+1} = \max(914{,}4; 853{,}3) = 914{,}4 \approx 915 \text{ MB}$$

A Figura 5.1 apresenta as séries temporais agregadas das métricas `ucpu`, `mr` e `mc` no servidor ZAB1 (o período foi limitado a uma semana para facilitar a visualização). Conforme discutido na Seção 2.3, os modelos de séries temporais buscam refletir o padrão exibido nos dados. Como pode ser constatado nos gráficos, as séries temporais das métricas não exibem de forma evidente – ao menos visualmente – um padrão que possa ser caracterizado como uma tendência ou uma sazonalidade. A característica exibida requer modelos mais complexos, pois quanto maior a dificuldade em identificar um padrão, maior a complexidade do modelo e maior o erro de previsão.

Figura 5.1: Amostra das métricas para servidor ZAB1 com dados agregados (intervalo de 1 h).



Fonte: O autor.

5.1.2 Tratamento dos Dados

A partir dos registros fornecidos pelo `sar` foram gerados arquivos de dados com *timestamps* no formato *AAAA-MM-DD HH:MM:SS* e os valores correspondentes para as métricas. Como a frequência do monitor é imprecisa (devido a flutuações de carga nos sistemas monitorados), para minimizar efeitos decorrentes de pequenas variações (na ordem de segundos) nos *timestamps*, estes foram arredondados para o minuto mais próximo (ou seja, *SS* é sempre zero). Ao final desta etapa observou-se que a métrica `steal` manteve-se constante e igual a zero nos dados coletados de todos os servidores, sendo desconsiderada da análise.

A análise das séries obtidas revelou algumas observações ausentes, provavelmente devido a falhas como, queda de energia, falha no sistema operacional ou falha no hardware. Como pretendia-se obter um período de medições minuto a minuto mínimo de três semanas (um total de $3 \times 7 \times 24 \times 60 = 30.240$ observações consecutivas), concluiu-se que não seria apropriado descartar períodos inteiros devido aos dados ausentes, mesmo porque não havia nos dados disponíveis nenhum período ininterrupto com tal número de observações. Existem técnicas ou procedimentos que podem ser usados para tratar dados ausentes. Um

procedimento é a *imputação de dados*, na qual dados ausentes são estimados com base nos dados existentes (RUBIN; LITTLE, 1987; HAIR, 2010). Algumas formas de imputação são: utilização da média das observações adjacentes e repetição do valor anterior.

Para preservar eventuais características sazonais, optou-se por imputar os dados ausentes com base nas observações disponíveis para o mesmo dia da semana e horário. Considerando a variabilidade encontrada nas observações, decidiu-se que os dados ausentes seriam imputados pela mediana das observações disponíveis. Assim, caso houvesse um dado ausente de uma segunda-feira às 15:15, por exemplo, essa observação seria substituída pela mediana das observações das demais segundas-feiras às 15:15. É importante registrar que não houve casos de mais de uma observação ausente para um dado dia da semana e horário. A Tabela 5.2 mostra o percentual de imputação dos dados que foi necessário para as métricas de cada servidor. O percentual de imputação foi o mesmo para todas as métricas de um mesmo servidor, tendo sido constatado um percentual de imputação máximo de 5,2% no servidor ZMU2.

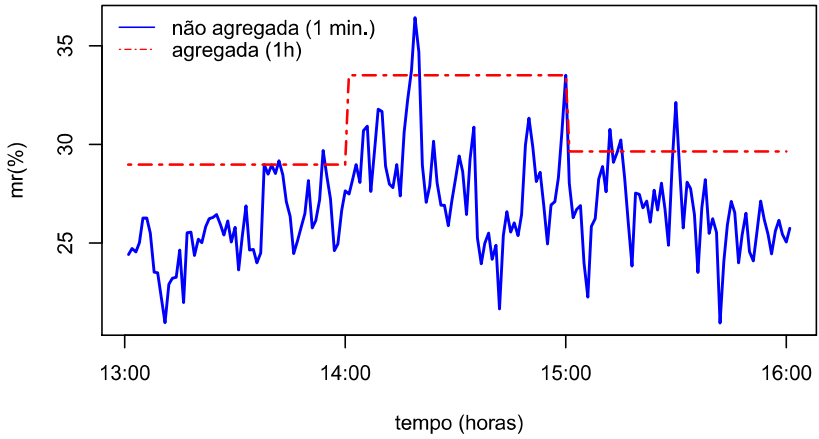
Tabela 5.2: Percentual de imputação (%) para cada servidor virtualizado.

ZAB1	ZMU1	ZMU2	ZMU3	ZMU4
0,0	1,26	5,2	0,29	0,09

Fonte: O autor.

Após o tratamento das observações ausentes, cada série foi agregada utilizando o 95º percentil, em períodos de uma hora. Assim, cada ponto agregado representa 60 pontos, resultando em 24 observações agregadas por dia. A Figura 5.2 mostra as observações originais e as agregadas da métrica de provisionamento `mr` durante um período de três horas. O gráfico mostra o efeito da agregação com o percentil de 95%. Visualmente é possível constatar que a maioria das observações não agregadas num período de uma hora são menores que a observação daquela hora. Ao agregar pelo percentil, cumpre-se o objetivo proposto de garantir que a maior parte das observações agregadas sejam maiores que a observações não agregadas a fim de evitar o subprovisionamento.

Figura 5.2: Efeito da agregação.



Fonte: O autor.

5.2 Ajuste de Modelos

A partir das séries agregadas obtidas na etapa anterior, os passos seguintes foram o ajuste de modelos e a obtenção das previsões. Para tal cada série de 22 dias foi dividida em:

- série de treinamento: primeiros 15 dias;
- série de testes: últimos 7 dias.

Os modelos foram ajustados usando a série de treinamento. As funções do pacote `forecast` foram utilizadas para obter modelos e previsões. Para os métodos Suavização Exponencial Simples (SES), Holt e Holt-Winters, uma mesma função (`ses`, `holt` e `hw`, de acordo com o método) foi utilizada tanto para ajustar modelos quanto para obter previsões. Para os métodos ETS e ARIMA foram utilizadas as funções `ets` e `auto.arima` para o ajuste de modelos, as quais retornam o melhor modelo para cada método; a função `forecast` aplicada a cada modelo ETS

ou ARIMA fornece previsões e intervalos de previsão. A tabela 5.3 resume as funções utilizadas.

Tabela 5.3: Funções do pacote `forecast` usadas na análise

método	funções usadas	
	ajuste de modelo	obtenção de previsões
SES	`ses`	`ses`
Holt	`holt`	`holt`
Holt-Winters	`hw`	`hw`
ETS	`ets`	`forecast`
ARIMA	`auto.arima`	`forecast`

Os métodos ETS e ARIMA podem produzir modelos com diferentes estruturas (com ou sem tendência, com ou sem sazonalidade). A Tabela 5.4 apresenta as estruturas dos modelos identificadas na etapa de ajuste. É possível constatar uma grande variação de modelos: para as 15 combinações disponíveis de servidor e métrica foram ajustados 10 modelos ETS e 15 modelos ARIMA distintos. Os métodos SES, Holt e Holt-Winters possuem estrutura fixa e não foram considerados nesta discussão.

Uma vez ajustados os modelos, foram obtidas as previsões um passo à frente (*one-step ahead*), utilizando as observações da série de testes. Para cada série foram calculadas 168 previsões, o mesmo comprimento da série de testes. Essas previsões foram então comparadas com as observações da série de testes, obtendo-se as métricas de diagnóstico MAE, POAP, SM e PAR descritas na Seção 4.2. Para o cálculo do MAE foi usada a função `accuracy` do pacote `forecast`, e para o cálculo das demais foram desenvolvidas funções no R.

5.3 Análise dos Resultados

A análise dos resultados está em duas partes: Análise Visual na Seção 5.3.1 e Análise das Métricas de Diagnóstico na Seção 5.3.2.

Tabela 5.4: Modelos obtidos pelos métodos ETS e ARIMA

Métrica	métodos ETS	ARIMA
ucpu	(M, N, M) / (ZAB1, ZMU1, ZMU2) (A,N,A) / (ZMU3) (M,Ad,M) / (ZMU4)	$(0,1,0)(1,0,0)_{24}$ / (ZAB1) $(1,1,1)(1,0,0)_{24}$ / (ZMU1) $(1,0,1)(0,0,0)_{24}$ / (ZMU2) $(4,0,2)(2,0,0)_{24}$ / (ZMU3) $(1,0,1)(2,0,0)_{24}$ / (ZMU4)
mr	(A,N,A) / (ZAB1, ZMU2) (M,N,M) / (ZMU1) (M,Ad,M) / (ZMU3, ZMU4)	$(1,1,1)(2,0,0)_{24}$ / (ZAB1) $(1,1,2)(0,0,1)_{24}$ / (ZMU1) $(2,1,3)(0,0,2)_{24}$ / (ZMU2) $(3,0,2)(2,0,0)_{24}$ / (ZMU3) $(0,1,1)(2,0,0)_{24}$ / (ZMU4)
mc	(A,N,A) / (ZAB1) (M,N,M) / (ZMU1, ZMU4) (M,N,A) / (ZMU2) (M,Ad,M) / (ZMU3)	$(2,0,5)(2,0,0)_{24}$ / (ZAB1) $(1,1,1)(0,0,1)_{24}$ / (ZMU1) $(1,1,1)(0,0,2)_{24}$ / (ZMU2) $(2,0,1)(2,0,0)_{24}$ / (ZMU3) $(0,1,0)(0,0,0)_{24}$ / (ZMU4)

Fonte: O autor.

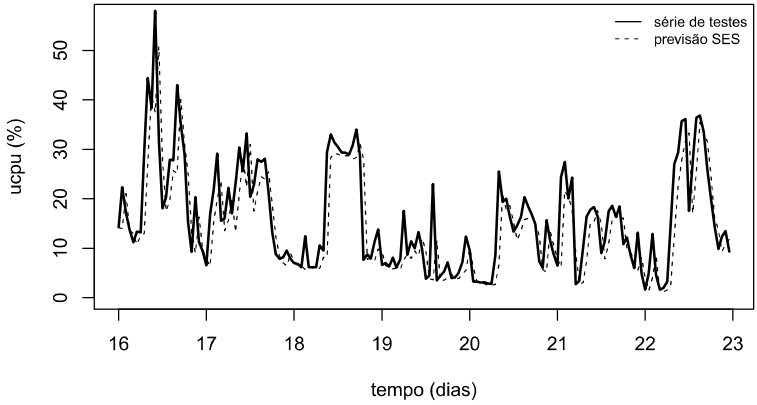
5.3.1 Análise Visual

A sobreposição das curvas das previsões de um método com a série de testes ajuda a compreender o comportamento do método e a identificar características que podem ajudar na escolha de um método.

Na Figura 5.3 e na Figura 5.4 foram plotadas as observações da série de testes da métrica ucpu e as previsões um passo à frente obtidas através dos métodos SES e Holt, respectivamente; os dados se referem ao servidor ZAB1. O servidor ZAB1 foi escolhido ao acaso para mostrar a análise visual, mas poderia ter sido escolhido qualquer um dos servidores sem prejuízo da análise. As previsões destes métodos apresentaram um problema em comum no conjunto de dados analisado: há um atraso visível entre a previsão e a observação. Se esta característica for significativa, se refletirá nas métricas de diagnóstico. A mesma característica é observada no método ARIMA da Figura 5.7, porém com menor frequência. O método ETS (Figura 5.6) apresenta nos picos e nos vales uma

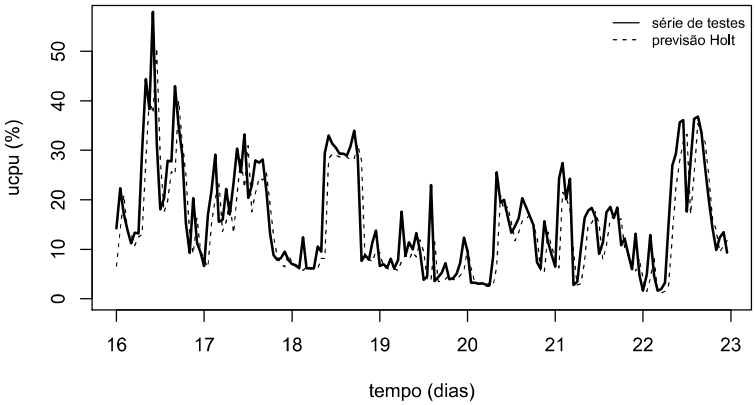
diferença em relação à observação mais significativa quando comparada com a diferença dos demais métodos. O método Holt-Winters (Figura 5.5) apresenta um atraso menor em relação aos demais métodos. A diferença entre vales e picos é menor que o método ETS, mas maior que SES, Holt e ARIMA.

Figura 5.3: Série de testes e previsões SES para métrica `ucpu` do servidor ZAB1.



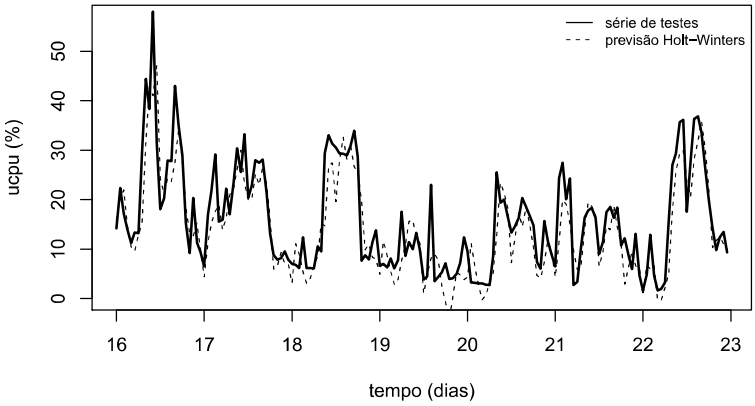
Fonte: O autor.

Figura 5.4: Série de testes e previsões Holt para métrica `ucpu` do servidor ZAB1.



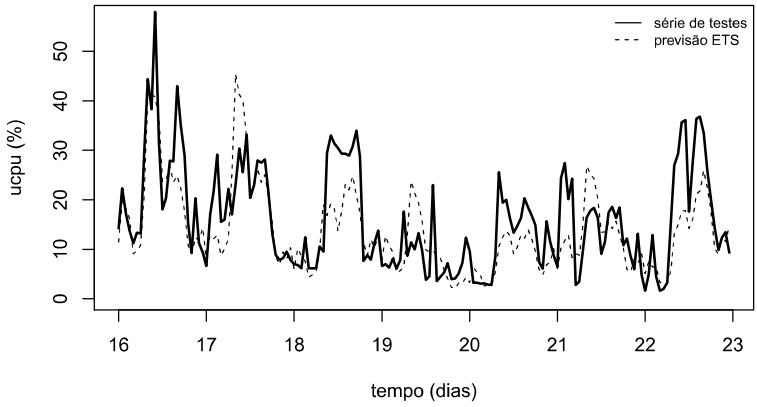
Fonte: O autor.

Figura 5.5: Série de testes e previsões Holt-Winters para métrica `ucpu` do servidor ZAB1.



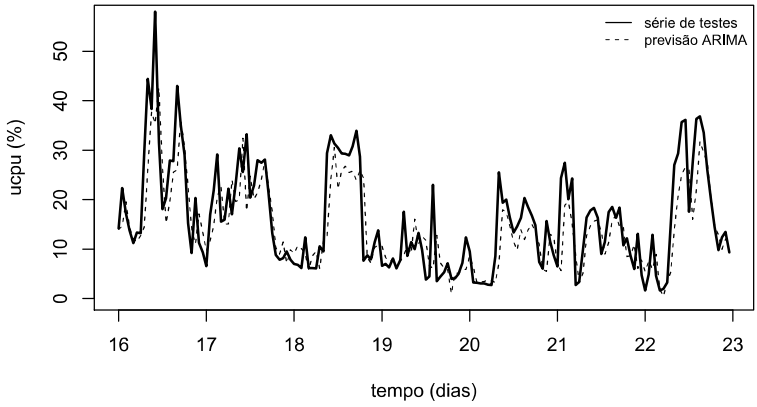
Fonte: O autor.

Figura 5.6: Série de testes e previsões ETS para métrica `ucpu` do servidor ZAB1.



Fonte: O autor.

Figura 5.7: Série de testes e previsões ARIMA para métrica `ucpu` do servidor ZAB1.



Fonte: O autor.

5.3.2 Análise das Métricas de Diagnóstico

Para a análise das métricas de diagnóstico de um servidor específico também foi escolhido o servidor ZAB1. Poderia ter sido escolhido outro servidor sem qualquer prejuízo ao processo de análise, mesmo considerando-se que os resultados são diferentes. A primeira métrica a ser analisada é o MAE, que quantifica o erro estatístico de previsão; os valores de MAE para ZAB1 são mostrados na Tabela 5.5. Os melhores resultados foram obtidos com os métodos ETS (para as métricas `ucpu` e `mc`) e Holt-Winters (para a métrica `mr`), e os piores resultados com os métodos Holt e SES. No geral, para ZAB1 o MAE foi baixo para `mr` e `mc`, mas nem tanto para `ucpu` (pois os valores estão próximos da média).

Tabela 5.5: Métrica de diagnóstico MAE para o servidor ZAB1

Método	`ucpu`	`mr`	`mc`
SES	4,78	0,14	0,65
Holt	5,11	0,14	0,75
Holt-Winters	1,25	0,10	0,44
ETS	1,12	0,11	0,42
ARIMA	1,83	0,13	0,45
	mín=0,23%	mín=4,16%	mín=8,02%
	média=2,03%	média=4,70%	média=11,86%
	max=19,86%	máx=5,35%	máx=13,20%

Fonte: O autor.

Para cada servidor e cada métrica de desempenho foi feito um ranking dos métodos de acordo com o MAE; o valor 1 indica o melhor método (menor MAE), e o valor 5 o pior método. A Tabela 5.6 apresenta o ranking médio dos métodos considerando todos os servidores, que é determinado pela média dos rankings de cada servidor. O método ETS foi o melhor, no geral, e os piores foram SES, Holt e ARIMA, com Holt-Winters em uma posição intermediária.

Conforme discutido na Seção 4.2, o MAE atribui o mesmo peso para subestimativas e superestimativas, mas no provisionamento de recursos esses dois tipos de erros têm consequências distintas. Neste con-

Tabela 5.6: Ranking médio dos métodos para métrica de diagnóstico MAE para ucpu, mr e mc considerando todos os servidores.

método	métricas		
	ucpu	mr	mc
SES	3,6	3,6	3,4
Holt	4,2	3,2	3,8
Holt-Winters	1,6	2,6	3,0
ETS	1,4	1,4	1,0
ARIMA	4,2	3,6	3,4

Fonte: O autor.

texto, as subestimativas são consideradas mais importantes porque um provisionamento que utilize uma subestimativa é, na prática, um subprovisionamento, o que prejudica o desempenho. Por outro lado, um provisionamento que utiliza uma superestimativa é, na prática, um superprovisionamento, o que causa gastos desnecessários com recursos. As métricas POAP e SM permitem avaliar as subestimativas; a primeira mensura a frequência de ocorrência de subestimativas, e a segunda a magnitude dessas ocorrências. Os valores das métricas para o servidor ZAB1 são mostrados na Tabela 5.7. Como as métricas de desempenho são expressas como porcentagens, os valores de SM estão em pontos percentuais (p.p.); isso significa, por exemplo, que uma observação de 20% para uma previsão de 15% corresponde a uma subestimativa de 5 p.p. Para a métrica ucpu, os valores de POAP foram baixos ($POAP_{max}$ = 4,09%), mas as subestimativas chegaram a 10,79 p.p. para os métodos SES e Holt. As métricas mr e mc apresentaram mais subestimativas, mas de pequena magnitude (SM_{max} = 1,05 p.p.). Essa inversão de tendência indica que, para o servidor ZAB1, os modelos de previsão para a métrica de CPU geraram menos subestimativas do que os modelos para as métricas de memória, mas estes tiveram erros menos significativos. Percebe-se ainda que os resultados para mc são similares aos resultados para mr, o que era esperado por se tratarem de métricas distintas para o mesmo recurso, porém relacionadas. Os melhores e piores valores de POAP e SM para as diferentes métricas foram obtidos com métodos variados, não havendo

nenhum que tenha se destacado.

A avaliação conjunta de POAP e SM oferece conclusões mais ricas do que análises isoladas. Por exemplo, embora o método ETS tenha obtido para `ucpu` um POAP igual ao dobro do daquele obtido pelo método de Holt, as subestimativas do ETS foram 8,5 p.p. inferiores. Como um déficit de 2,23 p.p. de CPU tende a causar um impacto pouco perceptível no desempenho, neste caso as subestimativas do ETS têm menos relevância que as do Holt, e o primeiro método pode ser considerado melhor que o segundo.

Tabela 5.7: Métricas de diagnóstico POAP (%) e SM (p.p.) para o servidor ZAB1

método	métricas					
	`ucpu`		`mr`		`mc`	
	POAP	SM	POAP	SM	POAP	SM
SES	1,11	10,79	9,59	0,12	7,65	0,92
Holt	1,09	10,74	9,65	0,13	6,80	1,05
Holt-Winters	1,83	2,80	8,92	0,09	6,04	0,90
ETS	2,18	2,23	8,80	0,09	6,92	0,82
ARIMA	4,09	2,30	7,99	0,12	12,11	0,40

Fonte: O autor.

Os resultados para todas as métricas e todos os servidores podem ser encontrados no Apêndice A, e a eles pode ser aplicada a mesma análise realizada para o servidor ZAB1. Ao invés de repetir a análise para cada servidor, optou-se por fazer uma análise global dos resultados. A métrica de diagnóstico MAE não é discutida na análise global porque é um valor absoluto, que deve ser avaliado em relação ao valor da métrica de provisionamento, e que varia de servidor para servidor.

Para mostrar uma visão mais geral do conjunto de servidores, os valores mínimos e máximos observados para a métrica POAP para cada método e para cada métrica são apresentados, respectivamente, nas Tabelas 5.8 e 5.9. Analisando essas tabelas, é possível constatar que:

- todos os métodos conseguiram obter valores baixos de POAP (mínimo entre 1,09% e 2,18%) para ao menos uma combinação métrica/servidor;
- o método ETS obteve POAP máximo superior ao dos demais métodos (entre 19% e 21%) para todas as métricas;
- as previsões de `ucpu` para o servidor ZAB1 e de `mc` para os servidores ZAB1 e ZMU2 obtiveram os melhores resultados;
- o servidor ZMU1 obteve os piores resultados, particularmente para previsão de memória.

Tabela 5.8: POAP mínimo e máximo por método

método	POAP mínimo (%)	POAP máximo (%)
SES	1,11 (ucpu/ZAB1)	10,04 (mr/ZMU1)
Holt	1,09 (ucpu/ZAB1)	9,70 (mr/ZMU1)
Holt-Winters	1,83 (mc/ZAB1)	16,84 (mc/ZMU1)
ETS	2,18 (ucpu/ZAB1)	21,19 (mc/ZMU1)
ARIMA	1,93 (mc/ZMU2)	13,95 (mr/ZMU1)

Fonte: O autor.

Tabela 5.9: POAP mínimo e máximo por métrica

métrica	POAP mínimo (%)	POAP máximo (%)
ucpu	1,09 (Holt/ZAB1)	13,14 (ETS/ZMU1)
mr	3,99 (HW/ZMU4)	19,44 (ETS/ZMU3)
mc	1,91 (HW/ZMU2)	21,19 (ETS/ZMU1)

Fonte: O autor.

A Tabela 5.10 mostra para cada servidor o maior POAP encontrado (com a maior subestimativa observada para a métrica/método) e a maior subestimativa encontrada (com o POAP correspondente para a métrica/método). Tipicamente, o POAP é inversamente proporcional à subestimativa, e vice-versa. Por exemplo, para o servidor ZMU3, o POAP máximo foi de 19,44% (métrica `mr`, método ETS), mas a maior subestimativa para `mr`/ETS foi 0,83 p.p. A maior subestimativa, 66,59 p.p., foi ve-

rificada no servidor ZMU2 (métrica ucpu, método Holt), mas o POAP para ucpu/HW foi de 2,54%. As piores combinações foram no servidor ZMU1: POAP máximo de 21,19% com subestimativa máxima de 9,31 p.p., e subestimativa máxima de 18,67 p.p. com POAP de 11,03%.

Tabela 5.10: POAP e S máximos por servidor (POAP em %, S em p.p.)

servidor	POAP máximo	S máx	S máxima	POAP
ZAB1	12,11 (mc/ARIMA)	0,50	16,06 (ucpu/SES)	1,11
ZMU1	21,19 (mc/ETS)	9,31	18,67 (mc/ARIMA)	11,03
ZMU2	10,49 (ucpu/ETS)	16,31	66,59 (ucpu/Holt)	2,54
ZMU3	19,44 (mr/ETS)	0,83	24,57 (ucpu/HW)	5,37
ZMU4	10,66 (mr/ETS)	1,08	8,14 (ucpu/SES)	5,00

Fonte: O autor.

A Tabela 5.11 apresenta a média de POAP considerando todos os servidores. A última coluna representa a média de POAP para cada método, e a última linha a média de POAP para cada métrica de desempenho. Os métodos que tiveram melhor POAP foram Holt (5,62%) e SES (5,69%), e o método que teve o pior POAP foi ETS (10,83%). A ordem relativa dos métodos se mantém praticamente a mesma para todas as métricas de desempenho. A métrica ucpu obteve o menor percentual de subestimativas (POAP=4,93%), enquanto mr obteve o maior percentual (POAP=8,38%).

Tabela 5.11: POAP médio para todos os servidores (valores em %)

	métricas			
método	ucpu	mr	mc	média
SES	4,18	7,00	5,90	5,69
Holt	4,10	7,15	5,62	5,62
Holt-Winters	4,46	8,04	7,64	6,71
ETS	7,57	11,94	12,99	10,83
ARIMA	4,35	7,77	6,76	6,29
média	4,93	8,38	7,78	

Fonte: O autor.

A Tabela 5.12 apresenta a média de SM considerando todos os

servidores. O método que obteve as menores subestimativas foi ETS (3,75 p.p.), e os piores métodos foram SES (8,08 p.p.) e Holt (7,99 p.p.). Da mesma forma que ocorreu com POAP, a ordem relativa dos métodos manteve-se praticamente inalterada para todas as métricas de desempenho. A métrica `mc` teve as menores subestimativas (2,13 p.p., em média), ao passo que `ucpu` teve as maiores subestimativas (média de 14,67 p.p., chegando a 18,55 p.p. com SES). As subestimativas para as métricas de memória são pequenas, indicando um impacto pouco significativo no desempenho. As subestimativas de `ucpu`, por sua vez, devem afetar o desempenho de forma mais perceptível, embora ocorram com frequência relativamente baixa (inferior a 5%, na média). Os dados das Tabelas 5.11 e 5.12 estão ilustrados na Figura 5.8, de modo a facilitar a análise conjunta de POAP e SM. De forma geral, constata-se que o POAP dos modelos é inversamente proporcional à SM menor, e vice-versa, tal qual observado em ZAB1.

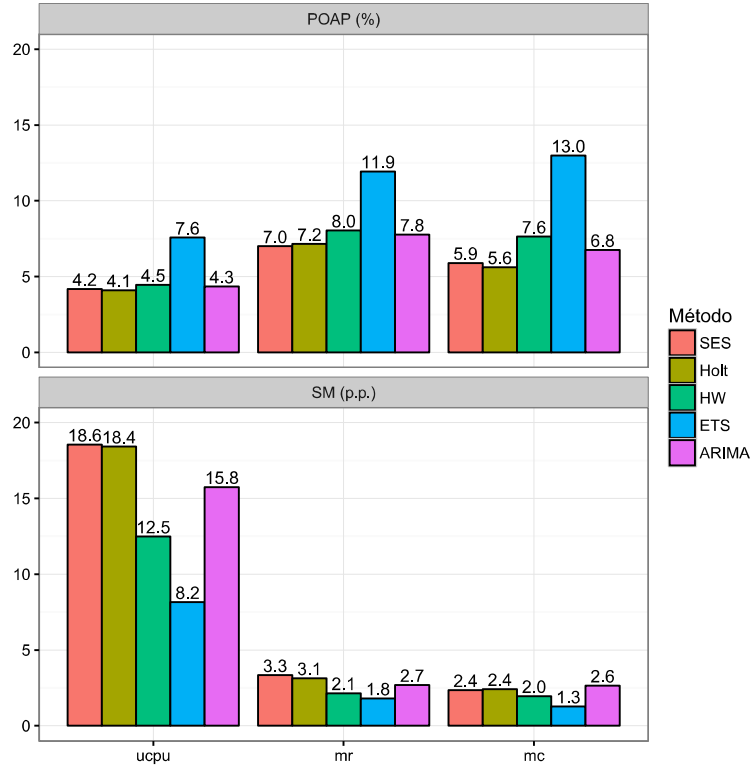
Tabela 5.12: Subestimativa média (SM) para todos os servidores (valores em p.p.)

	métricas			
método	`ucpu`	`mr`	`mc`	média
SES	18,55	3,34	2,35	8,08
Holt	18,42	3,14	2,42	7,99
Holt-Winters	12,49	2,15	1,96	5,53
ETS	8,16	1,81	1,28	3,75
ARIMA	15,75	2,69	2,65	7,03
média	14,67	2,63	2,13	

Fonte: O autor.

As métricas MAE, POAP e SM oferecem medidas dos erros de previsão fornecidos pelos diferentes métodos. Uma estimativa do custo relativo de provisionamento de cada método é fornecida pela métrica previsão acumulada relativa, PAR. A Tabela 5.13 mostra a média de PAR considerando todos os servidores. A métrica `ucpu` teve a maior diferença de PAR entre métodos, 0,20% para ETS e 31,52% para Holt, com média de 16,89%. Para as métricas de memória, as diferenças de PAR entre os

Figura 5.8: Média de POAP e SM para todos os servidores



Fonte: O autor.

métodos não passaram de 3,01%. Os métodos que levaram a uma maior previsão acumulada foram Holt ($\overline{\text{PAR}}$ = 12,02%) e SES ($\overline{\text{PAR}}$ = 10,89%), e o que gerou a menor demanda acumulada foi ETS, que produziu a menor PAR para 12 das 15 possíveis combinações de servidores e métricas, e que teve PAR máxima de 2,07% (métrica `mr` do servidor ZMU2). Portanto, pode-se afirmar que para o conjunto de servidores avaliados, o provisionamento com base nos modelos ETS teria o menor custo ao longo dos sete dias da série de testes, enquanto os provisionamentos com base nos modelos SES e Holt teriam os maiores custos.

Tabela 5.13: Previsão acumulada relativa (PAR) média para todos os servidores (valores em %)

	métricas			
método	ucpu	mr	mc	média
SES	28,51	1,51	2,66	10,89
Holt	31,52	1,53	3,01	12,02
Holt-Winters	8,72	2,01	2,45	4,40
ETS	0,20	0,43	0,00	0,21
ARIMA	15,49	1,79	2,42	6,57
média	16,89	1,46	2,11	

Fonte: O autor.

5.3.3 Discussão

Os resultados obtidos permitem avaliar a qualidade das previsões fornecidas pelos diferentes métodos investigados. As métricas de diagnóstico introduzidas no Capítulo 4 (POAP, SM e PAR) oferecem diferentes visões sobre o desempenho dos métodos. As duas primeiras quantificam as subestimativas encontradas, e a última possibilita a comparação dos custos de provisionamento associados a cada método. A ocorrência de erros é intrínseca ao processo de previsão: qualquer modelo de previsão comete erros. O importante é que esses erros sejam limitados em termos de frequência e magnitude. Nesse sentido, é possível afirmar que os métodos baseados em séries temporais investigados neste trabalho permitem obter estimativas da demanda de recursos úteis para fins de provisionamento.

Outras conclusões que podem ser extraídas dos resultados:

- As diferenças de configuração e de carga de trabalho entre os servidores monitorados – resumidas na Tabela 5.1, página 66 – são responsáveis por comportamentos distintos em termos de demanda de recursos, o que faz com que um método que apresente o melhor resultado para um servidor A possa não ser o melhor para um servidor B. De maneira análoga, os métodos oferecem resultados diferentes dependendo da métrica de provisionamento que está sendo prevista, mesmo conside-

rando um único servidor. Como consequência, é impossível eleger *a priori* um método de previsão como o melhor, sendo mais recomendável ajustar modelos distintos e avaliar a qualidade das previsões para cada combinação de servidor e métrica de provisionamento.

- Embora a metodologia proposta no Capítulo 4 pretendesse permitir a identificação de sazonalidade semanal e a avaliação experimental tenha sido conduzida usando séries de treinamento de 15 dias, tal sazonalidade não foi constatada nos modelos encontrados. A ordem sazonal[2] máxima encontrada usando ARIMA foi 2 (correspondente a 48 h), enquanto que a ordem para sazonalidade semanal seria 7, correspondente a $7 \times 24 = 168$ h. Para os métodos de suavização exponencial com sazonalidade (Holt-Winters e ETS), o período sazonal encontrado foi $m = 24$, e não $m = 168$; observa-se que, como 168 é múltiplo de 24, a observação de uma semana anterior influencia a previsão, embora com um efeito bastante pequeno. Seria necessária uma investigação mais aprofundada para determinar se as séries realmente não apresentam sazonalidade semanal ou se seria necessário utilizar outros modelos, como o Fourier-ARIMA (HAIDU, 1987), por exemplo.

- As subestimativas produzidas pelos diferentes métodos não foram muito significativas, mesmo considerando o pior caso. Em geral, métodos com subestimativas mais frequentes produziram subestimativas de menor magnitude, e as subestimativas de maior magnitude foram compensadas por uma baixa frequência.

- As maiores subestimativas foram encontradas principalmente para a métrica de utilização de CPU, que é naturalmente mais volátil que a utilização de memória (mensurada por `mr` e `mc`). Tais subestimativas correspondem a picos de utilização de natureza aperiódica, que constituem um desafio para qualquer método de previsão.

[2] A ordem de um modelo de série temporal indica o número de observações passadas da variável incorporadas no modelo (MORETTIN; TOLOI, 2006). A ordem sazonal refere-se ao número de períodos sazonais incorporados no modelo.

- As métricas de diagnóstico propostas neste trabalho (POAP, SM e PAR) são complementares do ponto de vista qualitativo. Numericamente, SM e PAR exibiram forte correlação positiva (o coeficiente de correlação para os valores obtidos foi $r = 0{,}922$), enquanto POAP e SM mostraram uma correlação negativa moderada ($r = -0,683$).

- A escolha de um método está associada ao que o cliente da nuvem IaaS considera como prioritário em termos de provisionamento, desempenho ou custo. Caso a prioridade seja desempenho, a escolha deve ser guiada pela análise conjunta das métricas POAP e SM; caso a prioridade seja custo, a escolha deve levar em conta a minimização da métrica PAR. O MAE está associado ao erro estatístico de previsão, e não se revelou muito útil para avaliar o provisionamento.

5.4 Considerações do Capítulo

Neste capítulo, dados coletados de servidores virtualizados em ambiente de produção foram usados na avaliação da metodologia proposta no Capítulo 4. As variações de configuração e carga de trabalho dos diferentes servidores levaram ao ajuste de 15 modelos distintos usando cada um dos cinco métodos investigados. Os 75 modelos foram avaliados com base em quatro métricas de diagnóstico, três das quais introduzidas neste trabalho.

Os resultados evidenciam que a aplicação da metodologia proposta conduz a modelos de séries temporais que geram previsões de demanda de recursos adequadas para provisionamento de capacidade em nuvens IaaS. Mostrou-se ainda que as métricas de diagnóstico criadas neste trabalho – POAP, SM e PAR – permitem quantificar as subestimativas produzidas pelos modelos e o custo relativo de provisionamento, possibilitando avaliar os modelos sob o ponto de vista de desempenho ou de custo.

6 CONCLUSÃO

As nuvens IaaS trouxeram um novo paradigma para o gerenciamento de infraestruturas computacionais. Ao invés de adquirirem e operarem suas próprias infraestruturas, as organizações podem alugar recursos virtuais (processador, memória, rede, disco) de provedores de nuvem, que se responsabilizam pelo gerenciamento do hardware. Um conceito chave da computação em nuvem é a elasticidade, que permite que um cliente possa aumentar ou diminuir a capacidade de seus recursos virtuais dinamicamente. Em termos econômicos, a infraestrutura computacional deixa de ser um custo fixo, com um investimento inicial muitas vezes vultoso, e passa a ser um custo variável, em função do consumo de recursos.

Apesar dos avanços ocorridos nos últimos anos, a elasticidade ainda não atingiu um estágio que permita que os clientes sejam tarifados exatamente pelos recursos consumidos. Na prática, o que ocorre hoje é que os clientes pagam pelos recursos alocados, independente de sua utilização, e essa alocação só pode ser ajustada a intervalos fixos, tipicamente de uma hora. Assim, é importante que um cliente seja capaz de provisionar corretamente a capacidade de seus recursos virtuais. Um subprovisionamento de recursos afeta o desempenho das aplicações e pode trazer sérias consequências financeiras (com perda de clientes ou descumprimento de objetivos previstos em contrato, por exemplo), ao passo que um superprovisionamento gera capacidade ociosa e, consequentemente, custos desnecessários. Alguns provedores IaaS oferecem mecanismos para ajuste dos recursos alocados de forma reativa, ou seja, depois que um problema for detectado. Antecipar o provisionamento adequado de recursos requer um conhecimento especializado do qual poucos clientes de nuvem dispõem.

O objetivo desta dissertação era investigar o uso de séries temporais para previsão da demanda de recursos em sistemas virtualizados,

com o propósito de encontrar modelos que pudessem ser usados para provisionamento proativo de recursos. Para isso, foi proposta uma metodologia para obter modelos de séries temporais a partir de dados de monitoração de recursos. Foram também introduzidas métricas para diagnóstico dos modelos obtidos sob a perspectiva do provisionamento, que permitem quantificar as subestimativas produzidas por um modelo e comparar o seu custo com outros modelos.

A metodologia proposta foi avaliada usando dados de cinco servidores virtualizados em produção, cinco diferentes métodos de previsão e três métricas de provisionamento, uma de processador e duas de memória. A diversidade de configurações e cargas de trabalho desses servidores evidenciou a necessidade de ajustar modelos específicos para cada servidor e métrica de desempenho de interesse. Os resultados mostraram que os modelos obtidos seguindo a metodologia proposta podem ser usados para provisionar recursos de forma proativa. As métricas de diagnóstico criadas para avaliar os modelos de previsão oferecem diferentes critérios para que um cliente possa selecionar um modelo priorizando o desempenho ou o custo, conforme sua conveniência.

Diversas linhas podem ser seguidas para a continuidade deste trabalho. Uma delas seria aprofundar a investigação da sazonalidade semanal nos dados, para confirmar ou não a presença dessa característica e buscar métodos alternativos de modelagem. Uma segunda linha seria avaliar experimentalmente o desempenho e o custo decorrentes de efetivamente provisionar os recursos segundo as previsões fornecidas pelos modelos. Em terceiro lugar, poderia ser investigada a frequência com que os modelos devem ser reajustados, uma vez que modelos de previsão de demanda de recursos devem ser atualizados para acompanhar a evolução dos sistemas e cargas de trabalho. Finalmente, a previsão de demanda de longo prazo para previsão de capacidade é uma área que também poderia ser explorada sob a perspectiva de um cliente de nuvem IaaS.

REFERÊNCIAS BIBLIOGRÁFICAS

ABDELZAHER, T. et al. Introduction to control theory and its application to computing systems. **Performance Modeling and Engineering**, Springer, p. 185–215, 2008.

ACETO, G. et al. Cloud monitoring: A survey. **Computer Networks**, Elsevier, v. 57, n. 9, p. 2093–2115, 2013.

BOX, G. E. P.; JENKINS, G. M.; REINSEL, G. C. **Time Series Analysis: Forecasting and Control**. 4th. ed. [S.l.]: John Wiley & Sons, 2008. ISBN 978-0-470-27284-8.

CHATFIELD, C. **Time-Series Forecasting**. 1st. ed. [S.l.]: Chapman and Hall/CRC, 2000. ISBN 1-58488-063-5.

CISCO. **Cisco Global Cloud Index: Forecast and Methodology, 2013-2018**. 2014. <http://www.cisco.com/c/en/us/solutions/collateral/service-provider/global-cloud-index-gci/Cloud_Index_White_Paper.pdf>.

ENGELBRECHT, H.; GREUNEN, M. van. Forecasting methods for cloud hosted resources, a comparison. In: IEEE. **Network and Service Management (CNSM), 2015 11th International Conference on**. [S.l.], 2015. p. 29–35.

FANG, W. et al. RPPS: A novel resource prediction and provisioning scheme in cloud data center. In: IEEE. **Services Computing (SCC), 2012 IEEE Ninth International Conference on**. [S.l.], 2012. p. 609–616.

GIL, A. C. **Como Elaborar Projetos de Pesquisa**. 4a. ed. [S.l.]: Atlas, 2002. ISBN 85-224-3169-8.

GODARD, S. **SYSSTAT**. 2016. Disponível *online* em <http://sebastien.godard.pagesperso-orange.fr/>.

GONG, Z.; GU, X.; WILKES, J. PRESS: Predictive elastic resource scaling for cloud systems. In: IEEE. **Network and Service Management (CNSM), 2010 International Conference on**. [S.l.], 2010. p. 9–16.

GREGG, B. **Systems Perfomance: Enterprise and Cloud**. [S.l.]: Prentice Hall, 2014. ISBN 978-0-13-339009-4.

HAIDU, I.; SERBRN, P.; SIMOTA, M. Fourier-ARIMA modelling of the multiannual flow variation. **LAHS Publ**, n. 168, p. 281–286, 1987.

HAIR, J. F. et al. **Multivariate Data Analysis**. 7th. ed. [S.l.]: Prentice Hall, 2010.

HUANG, J.; LI, C.; YU, J. Resource prediction based on double exponential smoothing in cloud computing. In: IEEE. **Consumer Electronics, Communications and Networks (CECNet), 2012 2nd International Conference on**. [S.l.], 2012. p. 2056–2060.

HYNDMAN, R.; KHANDAKAR, Y. Automatic time series forecasting: The forecast package for R. **Journal of Statistical Software**, v. 27, n. 1, p. 1–22, 2008. ISSN 1548-7660. Disponível em: <https://www.jstatsoft.org/index.php/jss/article/view/v027i03>.

HYNDMAN, R. J.; ATHANASOPOULOS, G. **Forecasting: Principles and Practice**. [S.l.]: Otexts, 2016. Disponível *online* em <https://www.otexts.org/fpp/>. ISBN 978-0-470-27284-8.

HYNDMAN, R. J.; KOEHLER, A. B. Another look at measures of forecast accuracy. **International Journal of Forecasting**, Elsevier, v. 22, n. 4, p. 679–688, 2006.

HYNDMAN, R. J. et al. A state space framework for automatic forecasting using exponential smoothing methods. **International Journal of Forecasting**, Elsevier Science B.V., v. 18, n. 3, p. 439–454, 2002.

HYNDMAN, R. J. et al. **Forecasting with Exponential Smoothing**. [S.l.]: Springer-Verlag Berlin Heidelberg, 2008.

KAUARK, F.; MANHãES, F. C.; MEDEIROS, C. H. **Metodologia da Pesquisa: Guia Prático**. [S.l.]: Via Litterarum, 2010. 25-27 p.

LAKATOS, E. M.; MARCONI, M. A. **Metodologia Científica**. 5th. ed. [S.l.]: Atlas, 2008. ISBN 978-85-224-4762-6.

LORIDO-BOTRAN, T.; MIGUEL-ALONSO, J.; LOZANO, J. A. A review of auto-scaling techniques for elastic applications in cloud environments. **Journal of Grid Computing**, Springer, p. 1–34, 2014.

MAKRIDAKIS, S.; HIBON, M. The M3-Competition: Results, conclusions and implications. **International journal of forecasting**, Elsevier, v. 16, n. 4, p. 451–476, 2000.

MARVASTI, M. A. **Quantifying Information Loss Through Data Aggregation**. 2011. Disponível em <https://http://citeseerx.ist.psu.edu/viewdoc/download;jsessionid=F068AC906CF1B41E34F3E5C95E3A2EEA?doi=10.1.1.204.1194&rep=rep1&type=pdf>.

MELL, P.; GRANCE, T. **The NIST Definition of Cloud Computing**. [S.l.], 2011. <http://csrc.nist.gov/publications/nistpubs/800-145/SP800-145.pdf>.

MONTGOMERY, D. C.; RUNGER, G. C. **Applied Statistics and Probability for Engineers**. 5th. ed. [S.l.]: John Wiley & Sons, 2011. ISBN 978-0-470-05304-1.

MORAIS, F. et al. Autoflex: Service agnostic auto-scaling framework for IaaS deployment models. In: IEEE. **13th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGrid)**. [S.l.], 2013. p. 42–49.

MORETTIN, P. A.; TOLOI, C. M. C. **Análise de Séries Temporais**. 2th. ed. [S.l.]: Edgard Blucher Ltda., 2006. ISBN 978-85-212-0389-6.

PAXSON, V.; ALLMAN, M.; CHU J. ANS SARGENT, M. Computing tcp's retransmission timer. **RFC Editor**, 2011.

PEARCE, M.; ZEADALLY, S.; HUNT, R. Virtualization: Issues, security threats, and solutions. **ACM Computing Surveys**, New York, NY, USA, v. 45, n. 2, p. 17:1–17:39, mar. 2013. ISSN 0360-0300.

PFITSCHER, R. J. **Diagnóstico do provisionamento de recursos para máquinas virtuais em nuvens IaaS**. Dissertação (Mestrado) — Universidade do Estado de Santa Catarina, 2014.

PFITSCHER, R. J.; PILLON, M. A.; OBELHEIRO, R. R. Diagnóstico do provisionamento de recursos para máquinas virtuais em nuvens IaaS. **31o. Simpósio Brasileiro de Redes de Computadores (SBRC)**, p. 599–612, 2013.

PFITSCHER, R. J.; PILLON, M. A.; OBELHEIRO, R. R. Customer-oriented diagnosis of memory provisioning for IaaS clouds. **ACM SIGOPS Operating Systems Review**, ACM, v. 48, n. 1, p. 2–10, 2014.

R FOUNDATION. **The R Project for Statistical Computing**. 2016. Disponível *online* em <http:/www.r-project.org>.

RUBIN, D.; LITTLE, R. J. A. **Statistical Analysis with Missing Data**. [S.l.]: John Wiley & Sons, 1987.

SULEIMAN, B. et al. On understanding the economics and elasticity challenges of deploying business applications on public cloud infrastructure. **Journal of Internet Services and Applications**, v. 3, p. 173–193, 2012.

VMWARE. **Server Virtualization with VMware vSphere**. 2016. Disponível *online* em <http://www.vmware.com/products/vsphere/>.

WEINGÄRTNER, R.; BRÄSCHER, G. B.; WESTPHALL, C. B. Cloud resource management: a survey on forecasting and profiling models. **Journal of Network and Computer Applications**, Elsevier, 2014.

WOOD, T. et al. Sandpiper: Black-box and gray-box resource management for virtual machines. **Computer Networks**, Elsevier, v. 53, n. 17, p. 2923–2938, 2009.

Apêndices

APÊNDICE A – RESULTADOS COMPLETOS PARA TODOS OS SERVIDORES

A.1 Servidor ZAB1

Tabela A.1: Métrica de diagnóstico MAE para o servidor ZAB1

ZAB1	Métrica de Provisionamento		
Método	ucpu	mr	mc
SES	4,78	0,14	0,65
Holt	5,11	0,14	0,75
HW	1,25	0,10	0,44
ETS	1,12	0,11	0,42
ARIMA	1,83	0,13	0,45
	mín.=0,23%	mín.=4,16%	mín.=8,02%
	méd.=2,03%	méd.=4,70%	méd.=11,86%
	máx.=19,86%	máx.=5,35%	máx.=13,20%

Fonte: O autor.

Tabela A.2: Métricas de diagnóstico POAP (%) e SM (p.p.) para o servidor ZAB1

ZAB1	Métricas					
	ucpu		mr		mc	
Método	POAP	SM	POAP	SM	POAP	SM
SES	1,11	10,79	9,59	0,12	7,65	0,92
Holt	1,09	10,74	9,65	0,13	6,80	1,05
HW	1,83	2,80	8,92	0,09	6,04	0,90
ETS	2,18	2,23	8,80	0,09	6,92	0,82
ARIMA	4,09	2,30	7,99	0,12	12,11	0,40

Fonte: O autor.

Tabela A.3: Subestimativa (p.p.) mínima e máxima da previsão para o servidor ZAB1

ZAB1	Métrica					
	ucpu		mr		mc	
Método	Mín.	Máx.	Mín.	Máx.	Mín.	Máx.
SES	3,93	16,06	0,03	0,24	0,47	1,20
Holt	4,66	15,80	0,04	0,25	0,61	1,35
HW	0,89	6,49	0,05	0,17	0,80	1,00
ETS	0,85	4,90	0,05	0,15	0,72	0,93
ARIMA	1,25	4,57	0,04	0,24	0,30	0,50

Fonte: O autor.

Tabela A.4: Métrica de diagnóstico PAR (%) para o servidor ZAB1

ZAB1	Método				
Métrica	SES	Holt	HW	ETS	ARIMA
ucpu	83,08	95,90	5,42	0,00	6,05
mr	0,33	0,28	0,00	0,1	0,38
mc	1,42	2,26	0,19	0,00	0,34

Fonte: O autor.

A.2 Servidor ZMU1

Tabela A.5: Métrica de diagnóstico MAE para o servidor ZMU1

MAE	Métrica de Provisionamento		
Método	ucpu	mr	mc
SES	6,06	3,92	5,14
Holt	6,28	4,14	5,29
HW	5,27	4,68	5,98
ETS	5,95	3,70	4,62
ARIMA	6,63	4,61	6,12
	mín.=1,63%	mín.=16,01%	mín.=10,55%
	méd.=16,11%	méd.=32,41%	méd.=31,75%
	máx.=58,01%	máx.=95,03%	máx.=123,44%

Fonte: O autor.

Tabela A.6: Métricas de diagnóstico POAP (%) e SM (p.p.) para o servidor ZMU1

ZMU1	Métricas					
	ucpu		mr		mc	
Método	POAP	SM	POAP	SM	POAP	SM
SES	6,79	5,74	10,04	5,41	8,77	8,66
Holt	6,27	5,76	9,70	5,35	8,29	8,89
HW	6,26	4,03	13,39	5,55	16,84	7,26
ETS	13,14	4,49	12,71	4,68	21,19	4,37
ARIMA	5,30	5,50	13,95	5,88	11,03	10,60

Fonte: O autor.

Tabela A.7: Métrica de diagnóstico SM (p.p.) para o servidor ZMU1

ZMU1	Método				
Métrica	SES	Holt	HW	ETS	ARIMA
ucpu	5,74	5,76	4,03	4,49	5,50
mr	5,41	5,35	5,55	4,68	5,88
mc	8,66	8,89	7,26	4,37	10,60

Fonte: O autor.

Tabela A.8: Subestimativa (p.p.) mínima e máxima da previsão para o servidor ZMU1

ZMU2	Métrica					
	ucpu		mr		mc	
Método	Mín.	Máx.	Mín.	Máx.	Mín.	Máx.
SES	1,34	13,56	0,43	9,84	0,54	16,92
Holt	1,34	13,35	0,50	9,70	0,48	17,22
HW	0,91	9,57	2,96	8,27	4,18	11,11
ETS	0,91	12,32	0,48	8,67	0,30	9,31
ARIMA	1,09	13,09	1,45	10,68	2,68	18,67

Fonte: O autor.

Tabela A.9: Métrica PAR (%) para o servidor ZMU1

ZMU1	Método				
Métrica	SES	Holt	HW	ETS	ARIMA
ucpu	9,81	12,31	11,32	0,00	16,84
mr	1,40	2,53	3,85	0,00	1,44
mc	4,24	5,03	5,51	0,00	4,17

Fonte: O autor.

A.3 Servidor ZMU2

Tabela A.10: Métrica de diagnóstico MAE para o servidor ZMU2

ZMU2	Métrica de Provisionamento		
Método	ucpu	mr	mc
SES	22,50	3,26	0,83
Holt	22,35	2,95	0,83
HW	15,34	3,23	0,58
ETS	13,07	3,06	0,29
ARIMA	22,88	3,52	0,80
	mín.=4,84%	mín.=8,40%	mín.=8,45%
	méd.=17,12%	méd.=31,36%	méd.=10,37%
	máx.=100,00%	máx.=46,09%	máx.=13,52%

Fonte: O autor.

Tabela A.11: Métricas de diagnóstico POAP (%) e SM (p.p.) para o servidor ZMU2

ZMU1	Métricas					
	ucpu		mr		mc	
Método	POAP	SM	POAP	SM	POAP	SM
SES	2,50	55,29	5,29	10,11	2,46	0,88
Holt	2,54	55,02	6,11	9,23	2,46	0,87
HW	3,05	33,44	7,58	4,09	1,91	0,44
ETS	10,49	10,73	8,12	3,65	7,72	0,38
ARIMA	2,56	50,19	6,72	6,41	1,93	0,90

Fonte: O autor.

Tabela A.12: Subestimativa (p.p.) mínima e máxima da previsão para o servidor ZMU2

ZMU2	Métrica					
	ucpu		mr		mc	
Método	Mín.	Máx.	Mín.	Máx.	Mín.	Máx.
SES	29,70	66,08	2,02	11,22	0,26	1,60
Holt	29,04	66,59	2,20	10,27	0,26	1,60
HW	15,38	41,03	2,15	4,87	0,23	0,78
ETS	5,00	16,31	1,80	4,46	0,08	1,00
ARIMA	28,36	62,09	3,09	7,30	0,44	1,45

Fonte: O autor.

Tabela A.13: Métrica PAR (%) para o servidor ZMU2

ZMU2	Método				
Métrica	SES	Holt	HW	ETS	ARIMA
ucpu	29,79	28,26	16,39	0,00	32,39
mr	1,15	0,00	2,50	2,07	2,62
mc	5,92	5,98	4,92	0,00	6,06

Fonte: O autor.

A.4 Servidor ZMU3

Tabela A.14: Métrica de diagnóstico MAE para o servidor ZMU3

ZMU3	Métrica de Provisionamento		
Método	ucpu	mr	mc
SES	24,75	0,85	0,97
Holt	25,52	0,84	0,94
HW	21,78	0,68	0,71
ETS	22,88	0,50	0,47
ARIMA	26,52	0,81	0,80
	mín.=15,54%	mín.=11,491%	mín.=47,04%
	méd.=63,33%	méd.=15,01%	méd.=50,57%
	máx.=100,00%	máx.=21,58%	máx.=58,66%

Fonte: O autor.

Tabela A.15: Métricas de diagnóstico POAP (%) e SM (p.p.) para o servidor ZMU3

ZMU3	Métricas					
	ucpu		mr		mc	
Método	POAP	SM	POAP	SM	POAP	SM
SES	5,50	17,39	5,80	0,55	5,46	0,51
Holt	5,33	17,21	6,00	0,49	5,96	0,38
HW	5,37	19,34	6,34	0,53	6,25	0,37
ETS	4,99	20,08	19,44	0,40	18,49	0,40
ARIMA	5,01	17,21	5,88	0,56	4,74	0,44

Fonte: O autor.

Tabela A.16: Subestimativa (p.p.) mínima e máxima da previsão para o servidor ZMU3

ZMU3	Métrica					
	ucpu		mr		mc	
Método	Mín.	Máx.	Mín.	Máx.	Mín.	Máx.
SES	14,59	19,67	0,05	1,11	0,04	1,24
Holt	14,67	19,16	0,04	0,96	0,04	0,92
HW	16,95	24,57	0,07	0,92	0,04	0,85
ETS	18,29	22,28	0,05	0,83	0,06	0,92
ARIMA	15,20	18,58	0,03	1,10	0,05	1,04

Fonte: O autor.

Tabela A.17: Métrica de diagnóstico PAR (%) para o servidor ZMU3

ZMU3	Método				
Métrica	SES	Holt	HW	ETS	ARIMA
ucpu	3,43	4,38	0,00	1,02	5,93
mr	3,95	4,07	3,09	0,00	3,78
mc	1,47	1,48	1,05	0,00	1,23

Fonte: O autor.

A.5 Servidor ZMU4

Tabela A.18: Métrica de diagnóstico MAE para o servidor ZMU4

ZMU4	Métrica de Provisionamento		
Método	ucpu	mr	mc
SES	3,43	0,16	0,25
Holt	3,47	0,17	0,27
HW	2,60	0,13	0,46
ETS	2,15	0,10	0,19
ARIMA	3,33	0,16	0,27
	mín.=3,30%	mín.=14,65%	mín.=43,65%
	méd.=11,61%	méd.=15,24%	méd.=49,52%
	máx.=33,29%	máx.=16,34%	máx.=51,55%

Fonte: O autor.

Tabela A.19: Métricas de diagnóstico POAP (%) e SM (p.p.) para o servidor ZMU4

ZMU4	Métricas					
	ucpu		mr		mc	
Método	POAP	SM	POAP	SM	POAP	SM
SES	5,00	3,52	4,27	0,50	5,18	0,80
Holt	5,26	3,37	4,29	0,50	4,58	0,89
HW	5,82	2,82	3,99	0,48	7,19	0,83
ETS	7,07	3,25	10,66	0,24	10,63	0,43
ARIMA	4,81	3,54	4,30	0,49	3,98	0,92

Fonte: O autor.

Tabela A.20: Subestimativa (p.p.) mínima e máxima da previsão para servidor ZMU4

ZMU4	Métrica					
	ucpu		mr		mc	
Método	Mín.	Máx.	Mín.	Máx.	Mín.	Máx.
SES	1,16	8,14	0,23	1,46	0,45	2,49
Holt	1,10	8,08	0,23	1,45	0,55	2,63
HW	1,01	7,43	0,17	1,22	0,54	2,05
ETS	0,76	7,85	0,05	1,08	0,14	1,48
ARIMA	1,05	8,13	0,22	1,30	0,49	2,81

Fonte: O autor.

Tabela A.21: Métrica PAR (%) para o servidor ZMU4

ZMU4	Método				
Métrica	SES	Holt	HW	ETS	ARIMA
ucpu	16,45	16,74	10,48	0,00	16,26
mr	0,74	0,76	0,61	0,00	0,73
mc	0,23	0,29	0,60	0,00	0,29

Fonte: O autor.

A.6 Resumo

Tabela A.22: Modelos obtidos pelos métodos ETS e ARIMA

Métrica	métodos ETS	ARIMA
ucpu	(M, N, M) / (ZAB1, ZMU1, ZMU2) (A,N,A) / (ZMU3) (M,Ad,M) / (ZMU4)	$(0,1,0)(1,0,0)_{24}$ / (ZAB1) $(1,1,1)(1,0,0)_{24}$ / (ZMU1) $(1,0,1)(0,0,0)_{24}$ / (ZMU2) $(4,0,2)(2,0,0)_{24}$ / (ZMU3) $(1,0,1)(2,0,0)_{24}$ / (ZMU4)
mr	(A,N,A) / (ZAB1, ZMU2) (M,N,M) / (ZMU1) (M,Ad,M) / (ZMU3, ZMU4)	$(1,1,1)(2,0,0)_{24}$ / (ZAB1) $(1,1,2)(0,0,1)_{24}$ / (ZMU1) $(2,1,3)(0,0,2)_{24}$ / (ZMU2) $(3,0,2)(2,0,0)_{24}$ / (ZMU3) $(0,1,1)(2,0,0)_{24}$ / (ZMU4)
mc	(A,N,A) / (ZAB1) (M,N,M) / (ZMU1, ZMU4) (M,N,A) / (ZMU2) (M,Ad,M) / (ZMU3)	$(2,0,5)(2,0,0)_{24}$ / (ZAB1) $(1,1,1)(0,0,1)_{24}$ / (ZMU1) $(1,1,1)(0,0,2)_{24}$ / (ZMU2) $(2,0,1)(2,0,0)_{24}$ / (ZMU3) $(0,1,0)(0,0,0)_{24}$ / (ZMU4)

Fonte: O autor.

Tabela A.23: Ranking médio dos métodos para métrica de diagnóstico MAE considerando todos os servidores

	Métricas		
Método	ucpu	mr	mc
SES	3,6	3,6	3,4
Holt	4,2	3,2	3,8
HW	1,6	2,6	3,0
ETS	1,4	1,4	1,0
ARIMA	4,2	3,6	3,4

Fonte: O autor.