

Análise de um Serviço Virtual de Armazenamento que Explora Classes de Objetos na Nuvem e Padrões de Acesso

Samuel de Oliveira Ribeiro¹, Glauber Dias Gonçalves¹, Francisco Airton Silva¹
Alex Borges Vieira², Jussara Marques de Almeida³

¹CSHNB - Universidade Federal do Piauí

²DCC - Universidade Federal de Juiz de Fora

³DCC - Universidade Federal de Minas Gerais

samueloliveira0014@gmail.com, {ggoncalves,faps}@ufpi.edu.br

alex.borges@ufjf.edu.br, jussara@dcc.ufmg.br

Resumo. *Um serviço virtual de armazenamento utiliza infraestruturas terceirizadas de nuvem para armazenar dados como objetos de diferentes classes e custos. Nesse artigo, investigamos um aspecto chave desse serviço: a predição da frequência de acessos aos dados para alocá-los em classes de armazenamento adequadas. Com base no padrão de acesso aos dados dos usuários, propomos um arcabouço que prediz, dinamicamente, as classes adequadas para armazenar dados (objetos) em um futuro próximo. Avaliamos a eficiência desse arcabouço através de simulações baseadas em traços de acesso a dados no Dropbox. Nossos resultados mostram alto desempenho em predições, que levam a uma economia de até 14% nos custos de armazenamento.*

Abstract. *A virtual storage service uses third-party cloud infrastructure to store data as objects of different classes and costs. In this article, we investigate a key aspect of this service: the prediction of the frequency of accesses to the data to allocate them in appropriate storage classes. predicting the frequency of data access to properly store data in different storage classes of the cloud infrastructure. Based on data access patterns of users, we propose a framework that dynamically predicts appropriate classes to store data (objects) in the near future. We evaluated the performance of this framework through trace-based data access simulations from Dropbox. Our results show high performance for predictions, which leads to savings of up to 14% in the storage costs.*

1. Introdução

Os serviços de armazenamento de dados em nuvem são uma alternativa ao tradicional armazenamento de dados em dispositivos locais. Através desses serviços, tarefas como *backup* com volume rapidamente escalável, replicação de dados em localidades diferentes, compartilhamentos e trabalhos colaborativos são facilitadas. As vantagens oferecidas aos usuários desse tipo de serviço, sejam eles domésticos ou corporativos, refletem as tendências crescimento desse setor. O índice global de computação em nuvem mais recente da Cisco prevê um volume de 1,3 ZB de dados armazenados na nuvem em 2022, um crescimento de 4,6 vezes comparado ao volume de 2016 [Cisco 2019]. Logo, líderes mundiais em tecnologia têm interesse em prover serviços de armazenamento em nuvem pessoal como Google Drive, Onedrive e iCloud. Mais ainda, provedores de infraestrutura como Amazon, Google Cloud e Microsoft oferecem kits de desenvolvimento –com

várias opções de acesso aos dados e preços— para que outras empresas possam criar seus próprios serviços de armazenamento em nuvem.

Em muitos casos, um provedor de serviço de armazenamento de dados em nuvem é apenas um serviço virtual, ou seja, o provedor de serviço aluga infraestrutura de um grande provedor de infraestrutura de nuvem, onde ele armazena os dados de seus usuários. O provedor de serviço então adiciona facilidades para uso, como uma interface amigável para computadores e dispositivos móveis, assim como opções de compartilhamento e versionamento de dados. Por exemplo, o Dropbox —um dos serviços mais populares de armazenamento em nuvem— utilizou a infraestrutura de nuvem da Amazon até 2016.¹ Observamos, nesses casos, as distinções entre o provedor de infraestrutura, o provedor de serviço ou aplicação e o usuário, que é o modelo clássico de computação em nuvem amplamente adotado por pesquisas na área [Josep et al. 2010].

Os principais provedores de infraestrutura em nuvem, por sua vez, oferecem diferentes classes para armazenamento de dados, baseado em objetos e variando seus custos por classe.² Um objeto é uma coleção lógica de bytes para representar arquivo com métodos bem definidos para acessos, com foco especial em leituras e compartilhamento [Mesnier et al. 2003]. Tipicamente, a classe de um objeto parte de uma analogia entre o seu nível de acesso e a sua “*temperatura*”: a classe Quente é para objetos frequentemente acessados pelo usuário; a classe Morna representa objetos com acessos infrequentes; a classe Fria, por sua vez, é para objetos antigos que não recebem mais acessos, se tornando inativos. O modelo de precificação dessas classes varia conforme o volume do objeto e o nível de acesso. Quanto maior a temperatura, maior o custo por unidade de volume (GB) e menor o custo por uma operação de acesso. Isso porque, os provedores de infraestrutura otimizam suas arquiteturas para cada classe de objetos, sabendo que objetos são muito acessados nos dias iniciais de sua existência (temperatura Quente), mas ao longo do tempo a frequência de acessos tende a diminuir (temperaturas Morna ou Fria) [Irie et al. 2018]. Os benefícios dessa otimização são então repassados em formas de custos variados aos provedores de serviço ou aplicações baseadas em nuvem.

Explorar classes de objetos na nuvem pode diminuir os custos dos provedores de serviço, em especial, um serviço de armazenamento virtual. Dada a organização em classes e a sua precificação, um serviço virtual de armazenamento deve encontrar as classes adequadas para objetos, de acordo com seus padrões de acesso, o mais rápido possível e assim, reduzir custos. Ao manter objetos infrequentes ou inativos classificados como Quente, perde-se a chance de economizar com armazenamento, dado que o custo por unidade de volume é menor nas demais classes. Mas, classificar objetos de forma equivocada pode, ao contrário do que se deseja, aumentar custos. Por exemplo, alocar na classe Morna um objeto que é acessado frequentemente, penaliza o custo final de armazenamento, pois o custo por acesso é alto para essa classe. O problema em questão é saber, antecipadamente, em qual classe alocar um determinado objeto de um usuário.

Neste artigo, propomos uma forma transparente para o provedor de serviço prever o padrão de acesso dos objetos criados/acessados por seus usuários, e assim, alocá-los apropriadamente em classes de armazenamento da infraestrutura de nuvem. Portanto, investigamos *em quais condições ou cenários, um provedor de serviço virtual de arma-*

¹<https://blogs.dropbox.com/tech/2016/05/inside-the-magic-pocket>

²<https://cloud.google.com/storage/pricing>, <https://aws.amazon.com/s3/pricing>

zenamento em nuvem pode obter lucratividade, isto é, pode reduzir custos, explorando os padrões de acessos dos usuários e as classes de objetos oferecidas por provedores de infraestrutura em nuvem.

Os trabalhos relacionados (Seção 2) em sua maioria focam em analisar padrões de acesso a dados para otimizar a infraestrutura em nuvem [Kaushik and Bhandarkar 2010, Muralidhar et al. 2014, Irie et al. 2018]. Esses trabalhos não exploram armazenamento a nível de serviço para usuário final, ao passo que um único trabalho com esse enfoque [Liu et al. 2019] não utiliza predição de acessos a dados. Nesse sentido, propomos um arcabouço para a predição da classe adequada de objetos em um serviço virtual de armazenamento em nuvem, integrado a um modelo geral de custo de objetos em provedores de infraestrutura (Seção 3). Avaliamos o desempenho do arcabouço em economizar custos do serviço em um cenário otimista e um cenário prático, através de simulações realistas, baseadas em traços de acessos de usuários coletados do Dropbox (Seção 4).

Nossos resultados apontam para as oportunidades e desafios no desenvolvimento de um serviço virtual de armazenamento em nuvem. Focamos nas classes de objetos Quente e Morna com o objetivo de economizar custos sem riscos de degradar a qualidade de serviço aos usuários. Primeiro, observamos analiticamente, via nosso modelo de custo, que o limiar de um acesso por mês compensa modificar a classe de objetos de Quente para Morna. Assim, aplicamos um classificador binário ao arcabouço para prever se há ou não acessos a objetos no futuro próximo. As simulações mostram que o arcabouço alcança predições com acuracidade, precisão e revocação acima de 80% e economia de custos de armazenamento de até 14%. Em suma, destacamos duas importantes contribuições nesse trabalho: (i) um arcabouço para prever classes de objetos, explorando características e os padrões de acesso dos usuários, visando otimizar os custos de um serviço virtual de armazenamento em nuvem; e (ii) um modelo para análise de custo de objetos, parametrizável de acordo os preços praticados por provedores de infraestrutura de armazenamento em nuvem.

2. Trabalhos Relacionados

Existem vários estudos sobre padrões de acesso a dados em sistemas de armazenamento. Uma linha de estudos dessa área foca no desenvolvimento de infraestruturas com requisitos mínimos de desempenho para armazenamento de dados em diferentes classes de acesso ou temperatura. Nesse sentido destacam-se [Kaushik and Bhandarkar 2010] e [Muralidhar et al. 2014]. No primeiro estudo, pesquisadores do Yahoo analisam três meses de traços de usuários na versão comercial do sistema Hadoop e propõem a variação GreenHDFS, que aloca arquivos infrequentes para servidores em modo de baixo consumo de energia, reduzindo gastos energéticos em 26%. No segundo estudo, os autores projetam uma infraestrutura de armazenamento intermediária (classe Morna) para dados que transitam entre acessos frequentes e infrequentes no Facebook.

Outros trabalhos tratam do dimensionamento de uma infraestrutura com suporte a duas classes de armazenamento: Quente e Fria. Em [Irie et al. 2018, Hsu et al. 2018] são propostos modelos para ordenar (“ranking”) arquivos do nível quente a frio, levando em consideração características como tipo do arquivo, número acessos e atividade dos usuários. Dada a ordenação de temperatura dos arquivos, os autores buscam o melhor compromisso para ajustar o tamanho do armazenamento quente (mais caro) de modo a

reduzir custos e também o impacto na qualidade do serviço aos usuários com acessos de maior latência a dados no armazenamento frio. Diferente desses trabalhos, nossa proposta de arcabouço e modelo de predição atua no nível de serviço, ou seja, aplicação baseada na infraestrutura da nuvem. Portanto, não consideramos limitações em tamanho de armazenamento, dado a infraestrutura altamente escalável da nuvem [Josep et al. 2010], focando assim no desempenho do modelo de predição para reduzir custos a nível de serviço.

Para nosso conhecimento, apenas um trabalho recente explora classes de armazenamento em nível de serviço de armazenamento de dados em nuvem. Em [Liu et al. 2019] é proposto um algoritmo *online* para modificar a classe do objeto ou transferi-lo para outro provedor com menor custo de armazenamento, dado o monitoramento de acessos a objetos em intervalos de tempo predefinidos. Essa é uma abordagem reativa, ou seja, pode haver várias modificações ou transferências de um mesmo objeto por período, o que tem forte impacto no custo de armazenamento. Diferente dessa, nossa abordagem baseia-se em predição para modificar a classe ou transferir um objeto uma única vez por período, quando necessário, tornando o custo de modificações ou transferências negligenciável. Nosso trabalho, portanto, é ortogonal a [Liu et al. 2019], cuja abordagem pode ser utilizada conjuntamente à nossa, enquanto não há dados suficientes para a predição.

Em vários trabalhos foram analisados o desempenho e caracterizados o comportamento de usuários em serviços pessoais de armazenamento na nuvem como Dropbox, Onedrive e Google Drive [Bocchi et al. 2015, Gracia-Tinedo et al. 2016, Gonçalves et al. 2016]. Os autores desses trabalhos sugerem várias otimizações para um serviço de armazenamento, mais especificamente com o foco em redução de custo com transferências e sincronizações de dados entre a nuvem e dispositivos de usuários em compartilhamentos. Contudo, em nenhum desses trabalhos os padrões de acesso de usuários foram analisados como uma oportunidade para reduzir o custo de um serviço virtual de armazenamento. Uma proposta para esse serviço foi apresentada em nosso trabalho anterior [Ribeiro et al. 2019], mas em uma versão preliminar, que não considera nem classes de objetos em nuvem e nem modelo de predição de acessos a dados.

3. Serviço Virtual de Armazenamento em Nuvem

Nesta seção, primeiro apresentamos o arcabouço para o nosso serviço virtual de armazenamento em nuvem (Seção 3.1). A seguir, discutimos o modelo de custo de armazenamento considerado (Seção 3.2).

3.1. Arcabouço para Serviço Virtual de Armazenamento em Nuvem

A Figura 1 apresenta uma visão geral do arcabouço para predizer classes de objetos, explorando características e os padrões de acesso dos usuários em um serviço virtual de armazenamento. Destacam-se os *usuários* e a *nuvem*, que são as entidades com as quais o serviço interage. Além disso, destacam-se três componentes do serviço: (1) *metadados*, (2) *predição* e (3) *controle*, com as setas indicando o funcionamento do serviço. A entidade usuário consiste em usuários domésticos/corporativos que buscam os benefícios de armazenamento em nuvem como compartilhamentos e backup. A entidade nuvem pode ser composta por um ou mais provedores de infraestrutura terceirizados, que o serviço virtual estabelece acordos de nível de serviço e preços para armazenar classes de objetos. A seguir, descrevemos cada componente do serviço seguindo o fluxo de funcionamento.

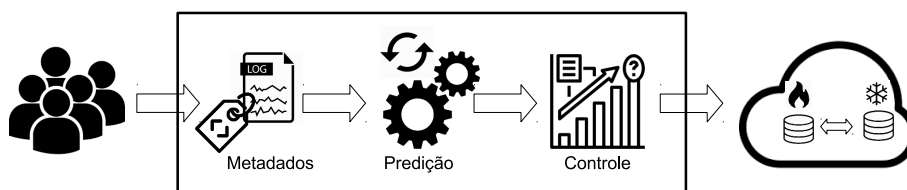


Figura 1. Arcabouço para prever classes de objetos, explorando características e padrões de acesso dos usuários em um serviço virtual de armazenamento.

O componente metadados registra as atividades dos usuários que são relevantes para caracterizar padrões de acesso a dados no serviço de armazenamento virtual. Exemplos dessas atividades são características do objeto como volume em bytes e a frequência de acessos, assim como características do usuário como a duração de sessões e quantidade de dispositivos. Portanto, esse componente deve coletar informações dos dados (metadados) gerenciados pelo serviço e dos usuários e dispositivos que acessam esses dados. Essas informações serão utilizadas posteriormente pelo serviço.

O componente predição atua como núcleo do arcabouço e utiliza os metadados para aprender características dos usuários e seus padrões de acesso, a fim de otimizar a operação do serviço. Esse componente tem a função mais importante do arcabouço em termos de custos e viabilidade do serviço de armazenamento, que é a predição de acessos a objetos. Para predições, o componente pode utilizar diferentes métodos estatísticos, ou algoritmos de inteligência artificial, ou até mesmo, um comitê de técnicas.

Especificamente, o modelo de predição deve indicar a classe, ou seja, a temperatura, em que os objetos devem ser definidos no futuro próximo, com objetivo de reduzir seus custos de armazenamento. Para isso, o modelo precisa ser ajustado em etapas de treino e teste, de acordo com uma *função de custo* para os objetos, que detalharemos na seção seguinte. Ambas as etapas ocorrem continuamente em períodos de tempo, tipicamente um mês e, utilizam as características dos objetos e dos usuários nos períodos passado e corrente. Predições incorretas podem aumentar o custo do armazenamento. Portanto, testes realizados periodicamente indicam o momento de aplicar efetivamente as predições de classes de objetos.

Finalmente, o componente controle atua como interface entre o serviço virtual de armazenamento e o provedor de infraestrutura na nuvem. Esse componente tem a função de escalar mudanças nas classes dos objetos na nuvem. O controle utiliza uma classe padrão, tipicamente a classe Quente, para objetos recém-criados. Quando os testes do módulo de predição indicam que o modelo está devidamente treinado, o controle utiliza essas predições, escalonando a data para modificar ou não a classe dos objetos.

3.2. Custo de Armazenamento

Nesta seção, propomos um modelo de custo de armazenamento de objetos em nuvem, aplicável a diferentes provedores de infraestrutura. Abstrairmos as especificidades de alguns provedores para focar em dois aspectos de maior impacto no custo de armazenamento: o volume e a frequência de acessos aos objetos [Hsu et al. 2018, Liu et al. 2019].

Primeiramente, mostramos como modelamos o custo de armazenamento praticado pelos maiores provedores de infraestrutura em função do volume e acessos aos objetos. Intuitivamente, usuários realizam vários acessos a um arquivo, aqui tratado como objeto,

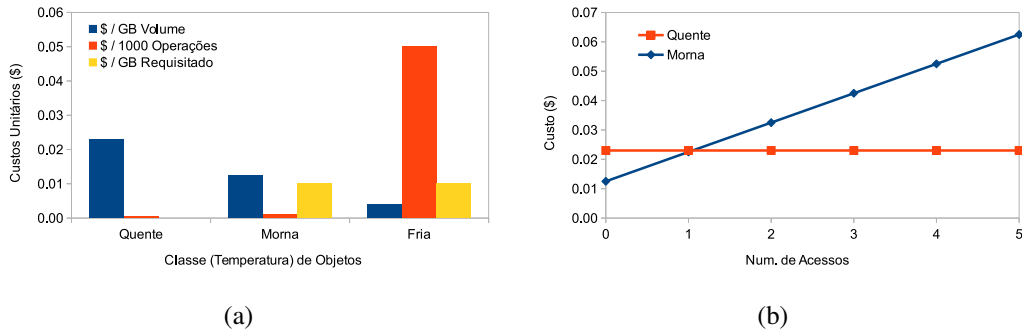


Figura 2. Custos para o armazenamento de classes de objetos na nuvem: (a) custos unitários por volume, operações e requisições; (b) custo de armazenamento em função do número de acessos (operações e requisições) para classes Quente e Morna. Valores extraídos do provedor Amazon modalidade S3.³

tão logo ele é criado e diminui o interesse pelo objeto com o passar do tempo, até que ele torna-se inativo. Assim, um objeto transita entre níveis decrescentes de acesso, representados usualmente pelos provedores de infraestrutura nas classes (ou temperaturas) Quente, Morna e Fria. O custo por classe varia conforme o volume do objeto e seus acessos: quanto maior a temperatura, maior o custo por volume e menor o custo por acesso. Esse custo é decomposto em custos unitários por volume (byte), operações (número de acessos) e requisições (bytes por acesso), exemplificados na Figura 2(a) com os custos do provedor Amazon na modalidade S3.³ Esse modelo também se aplica a outros grandes provedores de infraestrutura como Google e Microsoft.⁴ Todos esses provedores utilizam a mesma estrutura de classes e custos unitários, e as modificações em valores ocorridas ao longo do tempo mantêm o padrão mostrado na Figura 2(a).

Assim, propomos uma função de custo para armazenar objetos em diferentes classes, baseada no modelo de precificação acima descrito. O custo para armazenar o objeto i na classe j é dado por:

$$C_{ij}(x_i, y_i) = v_j \times x_i + o_j \times y_i + r_j \times x_i \times y_i, \quad (1)$$

onde x_i é o volume e y_i é a quantidade de acessos ao objeto i . Os custos unitários para armazenamento do objeto i na classe j são dados por: volume em GB v_j ; operação o_j , ou seja, custo por acesso;⁵ e requisição r_j , isto é, volume dos acessos em GB. Isso ocorre dentro de um período t , omitido para melhorar a clareza da equação.

O custo de armazenamento de um objeto $C_{ij}(x_i, y_i)$ pode ser reduzido, caso a classe do objeto seja definida de acordo à variação da sua quantidade de acessos ao longo do tempo. Na Figura 2(b), demonstramos esse custo analiticamente em função do número de acessos a um objeto. Nesse exemplo, assumimos um objeto com volume de 1 GB, e os custos unitários para as classes Quente e Morna mostrados na Figura 2(a) para o provedor Amazon na modalidade S3. Os custos de outros provedores, como Google e Microsoft, para modalidade e classes de objetos similares resultam em análises qualitativamente

³Simple Storage Service <https://aws.amazon.com/s3/pricing> (custos unitários em janeiro de 2020).

⁴<https://cloud.google.com/storage/pricing>, <https://azure.microsoft.com/pricing/details/storage/blobs>.

⁵Provedores de infraestrutura podem ter custos diferentes por leitura e inserção. Por simplicidade, utilizamos custo único para ambas operações, mas o modelo pode ser facilmente estendido para diferenciá-las.

equivalentes. Note que, para ambas as classes o custo de armazenamento cresce com o número de acessos, mas de forma visivelmente maior para a classe Morna. Podemos observar um limiar em número de acessos onde as curvas se encontram, ou seja, os custos se igualam, e a seguir modifica-se a classe com o menor custo. Para as parametrizações utilizadas nessa figura, esse limiar ocorre na marca de um acesso. Nota-se os menores custos antes e após o limiar respectivamente para as classes Morna e Quente. A mesma análise aplica-se também às combinações de duas classes com a classe Fria.

Utilizamos a nossa função de custo para obter esse limiar de número de acessos, que denominamos \hat{y}_i , assumindo que o volume do objeto é fixo.⁶ Assim como em [Liu et al. 2019], derivamos o limiar \hat{y}_i ao igualar os custos para o mesmo objeto em duas classes distintas a e b , isto é, $C_{ia}(x_i, \hat{y}_i) = C_{ib}(x_i, \hat{y}_i)$. A equação abaixo expressa o limiar \hat{y}_i a partir da nossa função de custo:

$$\hat{y}_i = \frac{x_i \times (v_a - v_b)}{(o_b - o_a) + x_i \times (r_b - r_a)}, \quad (2)$$

onde a classe a tem temperatura maior que a classe b para um volume fixo x_i . Os custos das unidades de volume (v), operação (o) e requisição (r) para cada classe são parametrizados de acordo com os preços adotados pelo provedor de infraestrutura.

A partir da Equação 2, pode-se observar as variáveis que impactam no limiar de números de acesso \hat{y}_i . Note que, o custo unitário por volume de armazenamento na classe de maior temperatura, v_a , pode aumentar o limiar de acessos. Por exemplo, ao aumentar o custo por unidade de volume na classe Quente (v_a) utilizado na Figura 2(b), o limiar de acessos \hat{y}_i entre as classes Quente e Morna aumenta, se tornando superior a um único acesso. Observa-se também, na Equação 2, que o volume dos objetos x_i reduz ou aumenta o impacto do custo unitário por operações no limiar \hat{y}_i . Por exemplo, objetos muito pequenos tendem ao limiar de zero acessos,⁷ ou seja, a classe Quente sempre tem o menor custo, ao passo que objetos muito grandes tendem a um limite para esse limiar, dado pela razão entre os custos unitários de volume e requisição $\frac{v_a - v_b}{r_b - r_a}$. Portanto, o limiar \hat{y}_i e os custos unitários do provedor de infraestrutura devem ser considerados na escolha do modelo de predição. Os custos utilizados na Figura 2(b), a exemplo, são adequados para modelos de predições binárias (i.e., zero ou mais acessos), discutidos na Seção 4.2.

4. Avaliação

Nesta seção, apresentamos a avaliação do arcabouço proposto para o serviço virtual de armazenamento em nuvem. Inicialmente, descrevemos o conjunto de dados reais utilizados para a avaliação. A seguir, apresentamos a metodologia e os resultados alcançados quanto à economia de custos de armazenamento potencial (ótima) e a economia de custos em cenários práticos com aplicação de um modelo de predição no arcabouço proposto.

4.1. Conjunto de Dados

Utilizamos traços reais de acessos a dados armazenados no Dropbox, coletados em nosso estudo anterior [Gonçalves et al. 2016] via dois pontos de presença (PoPs) de provedores

⁶Um objeto é imutável, e modificações em seu volume (operação de inserção) geram um novo objeto, com a opção do cliente manter versões anteriores do objeto ou descartá-las.

⁷Note a redução do numerador da Equação 2 com a redução do volume x_i .

Tabela 1. Sumário dos traços de acessos a dados do Dropbox.

Local	Dispositivos	Objetos	Acessos (x1000)	Período de Coleta
PoP-1	2300	10918	281	10/2013-04/2014
PoP-2	773	4292	364	07/2013-05/2014

de Internet residenciais, denominados PoP-1 e PoP-2.⁸ Os traços registram os acessos de usuários aos seus dados via esses PoPs, onde cada dado é um namespace do Dropbox, isto é, uma estrutura utilizada nesse serviço para identificar de forma única um arquivo do usuário (documento, áudio, imagem) ou um diretório. Logo, cada *namespace* é um objeto que o usuário armazena no serviço e posteriormente acessa. Os acessos são representados por identificadores anônimos (ID), que não oferecem dicas sobre a identidade dos usuários ou o conteúdo armazenado no serviço, mas nos permite analisar o padrão de acesso a dados no Dropbox. Dentre as informações sobre cada acesso de usuário, utilizamos para as nossas análises: a marca de tempo do acesso, o ID do objeto, o ID do dispositivo do usuário que fez o acesso e uma estimativa do volume do objeto. Esse volume é a soma dos bytes em *uploads* ou *downloads* por versão do objeto, representando uma estimativa conservadora (volume maior) para os objetos com múltiplas versões. Além dos acessos, os traços também registram eventos de *login* e *logout* no Dropbox que nos permite extrair mais características sobre usuários como tempos de sessões no serviço.

A Tabela 1 sumariza as informações dos traços utilizados para essa avaliação. Foram coletados pelo menos 6 meses de dados em cada PoP. No PoP-1, foram observados mais de 281 mil acessos e quase 11 mil objetos distintos, referentes a 2300 dispositivos. No PoP-2, foram observados mais de 364 mil acessos, porém com uma menor diversidade de objetos e dispositivos. Nas Figuras 3(a), 3(b) e 3(c) apresentamos as funções de distribuições acumuladas (CDF) das características mais importantes dos objetos observados. Para maiores informações, indicamos que o leitor acesse nosso estudo anterior [Gonçalves et al. 2016]. De maneira geral, os dois pontos de presença apresentam distribuições semelhantes. Os objetos armazenados têm tendência de serem pequenos. De fato, cerca de 60% dos objetos são menores que 100MB. Porém, cerca de 10% dos objetos têm mais de 1GB, o que pode exacerbar o custo de erros do modelo de predição no arcabouço proposto. Note que os objetos tendem a ser pouco acessados. Nesse caso, mais de 80% dos objetos não foram acessados mais de 10 vezes por mês, como evidencia a Figura 3(b). Por fim, o tempo de vida dos objetos tende a ser ligeiramente maior no PoP-1, como mostra a Figura 3(c). A maior atividade no PoP-1 pode significar menos oportunidades para utilizar classes Morna ou Fria em objetos e, por consequência, menor redução de custos de armazenamento.

4.2. Metodologia

Nossa avaliação foca nas classes de objetos Quente e Morna, visto que nesse trabalho buscamos reduzir custos do serviço de armazenamento virtual sem riscos de degradação de qualidade de serviço aos usuários. Reiteramos que, para essas classes, os maiores provedores de infraestrutura em nuvem mantêm a melhor qualidade de serviço, especificamente, o tempo de acesso a objetos.

Uma etapa importante da avaliação consiste na definição do modelo de predição do arcabouço. Neste trabalho, *propomos modelos de aprendizagem de máquina supervi-*

⁸Conjuntos de dados disponíveis em: <https://sites.google.com/a/ufpi.edu.br/traces>.

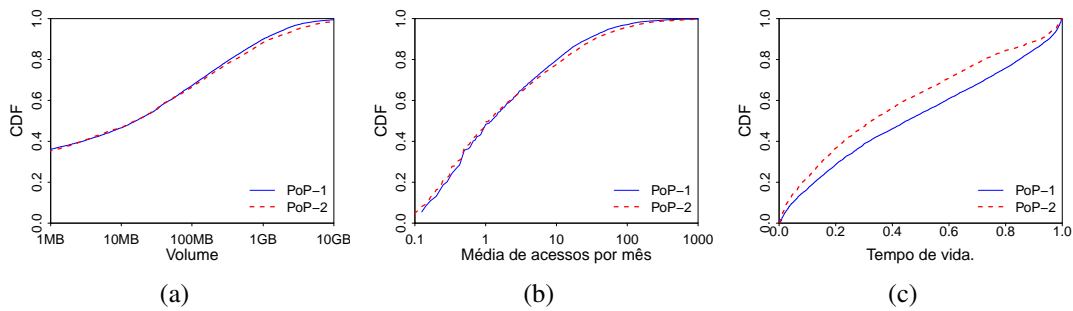


Figura 3. Características dos dados (objetos) Dropbox em funções de distribuições acumuladas (CDF): (a) volume nos objetos, (b) número médio de acessos por mês, e (c) tempo de vida do objeto relativo ao tempo do traço.

sionados para prever acessos futuros aos objetos de forma binária, ou seja, prever se o objeto terá ou não acesso no período seguinte. Essa escolha considera as nossas análises sobre o custo de armazenamento de objetos mostradas na Seção 3.2. Estendemos essas análises para objetos com diferentes volumes e observamos para a maioria dos objetos o limiar de um acesso, isto é $\hat{y}_i = 1$, ao qual modificar a classe de objetos de Quente para Morna reduz o custo de armazenamento. Portanto, em tese, é suficiente para o modelo prever se objetos terão acesso ou não no período seguinte, dado as características dos objetos e seus usuários.

Conduzimos simulações baseadas nos traços PoP-1 e PoP-2 para representar a aplicação do arcabouço proposto na Seção 3.1 realisticamente. Nas simulações, aplicamos uma metodologia de aprendizagem incremental no tempo, seccionando cada traço em períodos fixos t , que acumulam características dos objetos. Para cada período corrente t , o módulo de metadados coleta informações sobre os objetos e usuários do serviço até t . Por sua vez, o módulo de predição treina o modelo com informações do período anterior ($t - 1$) e o período corrente (t), e testa o desempenho do modelo em prever a classe dos objetos do período corrente no futuro próximo, que é o período seguinte ($t + 1$). O módulo controle aplica as predições aos objetos se os testes mostram desempenho satisfatório. Utilizamos períodos de 4 semanas, que é praticamente o período de pagamento de infraestrutura em nuvem (mensal). Assim, o primeiro teste requer 12 semanas (quase um trimestre), e a partir desse, decisões sobre o uso das predições podem ser avaliadas a cada teste, ou seja, períodos de 4 semanas.

As características dos objetos que utilizamos para treino e teste são: (1) o conjunto dos números de acessos semanais do objeto; (2) a média de duração de sessões do usuário, (3) a quantidade de objetos do usuário, (4) a média de acessos a objetos do usuário; e (5) a quantidade de objetos compartilhados do usuário, onde o usuário se refere ao(s) dono(s) do objeto. Por simplicidade, assumimos cada dispositivo como um usuário distinto. Todas as cinco características são calculadas para cada período de treino e teste simulados. Conduzimos as avaliações com diferentes classificadores da biblioteca de aprendizagem de máquina *Scikit-learn*⁹ na linguagem Python 3, utilizando suas configurações padrões. Contudo, balanceamos a quantidade de objetos nas classes Quente e Morna igualmente nos treinos, removendo objetos da classe Morna (dominante) aleatoriamente.

⁹<https://scikit-learn.org>

Para avaliar a economia com redução de custos de armazenamento utilizamos a métrica *r_{cs}* [Gonçalves et al. 2016], abreviação do inglês *relative cost savings*, dada por:

$$r_{cs_t} = \frac{cost_padrao_t - cost_modelo_t}{cost_padrao_t}, \quad (3)$$

onde *cost_padrao* é o custo do serviço virtual com todos os objetos em uma única classe (Quente ou Morna), que é o caso de não se utilizar nenhuma abordagem para reduzir custos. Por outro lado, *cost_modelo* é o custo do serviço com a aplicação do nosso arcabouço. Ambos os custos são calculados para todos os objetos avaliados no período corrente *t*.

4.3. Resultados

Analisamos o custo de armazenamento de objetos com arcabouço proposto para responder duas questões: (Q1) *Qual é o potencial máximo de economia que tal arcabouço pode explorar para armazenar objetos na nuvem, sem perdas da qualidade de serviço em tempos de acesso?* (Q2) *Qual é o percentual de economia que pode ser alcançado realisticamente ao tentar prever a classe ideal (menor custo) para objetos no futuro próximo?*

Potencial Máximo para Economia de Custos: Para responder Q1, calculamos o *r_{cs}*, considerando um modelo oráculo que sempre tem a predição correta para a classe do objeto no período seguinte. Ou seja, o modelo apresenta a economia de custos ótima. Isso nos dá um limite de ganhos potencial para economia de custos do arcabouço.

Inicialmente, nossa análise utiliza a classe Quente como padrão. Ou seja, comparamos a economia de manter objetos armazenados sempre na classe Quente à economia de custos ótima. Nesse caso, observamos que o *r_{cs}* no PoP-1 inicia com 18,69% no primeiro período de avaliação e alcança 25,85% após cinco períodos, devido ao crescimento da proporção de objetos que são mais adequados para a classe Morna. O mesmo ocorre para o traço PoP-2, onde o *r_{cs}* ótimo é 9,20% no primeiro período e 24,27% após nove períodos de avaliação, dado que esse traço é mais longo. O potencial de economia é maior para PoP-1, mesmo em um período menor de avaliação. De fato, o número de objetos analisados é maior nesse provedor, em comparação ao PoP-2 e, conforme esperado, quanto maior o número de predições com um modelo oráculo, maior é a economia. Tal economia seria relevante para um serviço virtual de armazenamento, pois é extraída de objetos ativos que podem ser acessados a qualquer momento pelo usuário com a mesma latência na classe Quente ou Morna, como determina os termos de nível de serviço dos provedores de infraestrutura atualmente. Portanto há um potencial para economia sem risco de reduzir qualidade do serviço. Caso a classe Morna seja o padrão para o *r_{cs}*, o potencial máximo de economia seria superestimado (*r_{cs}* superior a 97%) em ambos os locais, pois armazenar objetos sempre na classe Morna é a pior estratégia em termos de custo. Logo, mantemos a classe Quente como padrão para uma análise conservadora.

Considerações Práticas para Economia de Custos: Para responder Q2, consideramos o arcabouço proposto com o modelo de predição descrito em nossa metodologia para os últimos três períodos em cada local. Isso porque apenas nesses períodos acumulasse a quantidade de objetos suficiente (mostrados na Tabela 1) para treinar os modelos adequadamente. Inicialmente, discutimos o desempenho do modelo quanto às predições, a seguir, analisamos os reflexos dessas predições expresso em *r_{cs}*.

Tabela 2. Desempenho da predição e economia de custos relativa (*r_{cs}*).

<i>Local</i>	<i>Classificador</i>	<i>Acurácia</i>	<i>auc-roc</i>	<i>f1-score</i>	<i>Precisão</i>	<i>Revocação</i>	<i>r_{cs}</i>
Pop-1	SVM	82.70%	82.29%	76.44%	72.41%	80.96%	2.48%
	Rand Forest	86.39%	84.04%	79.55%	83.03%	76.36%	3.07%
Pop-2	SVM	81.20%	80.24%	70.23%	63.85%	90.43%	14.94%
	Rand Forest	83.50%	80.87%	72.04%	69.51%	89.66%	14.89%

Mostramos na Tabela 2 os dois modelos que obtiveram os melhores resultados, são eles os classificadores *Support Vector Machine* (SVM) e *Random Forest*.¹⁰ Avaliamos o desempenho das predições por meio de sua acurácia, *auc-roc*, *f1-score*, precisão e revocação, que são usuais para avaliar classificadores. Essas métricas estão apresentadas na Tabela 2. A acurácia indica a porcentagem de acertos (verdadeiros positivos e verdadeiros negativos) sobre o total de predições realizadas. A métrica *auc-roc* representa a porcentagem da área ocupada pela curva *roc* (variação da taxa de verdadeiros positivos com falsos positivos), e quanto maior essa porcentagem, melhor o modelo distingue classes. Percentuais próximos a 50% indicam que o modelo não é melhor que uma predição aleatória ou predição fixa em uma única classe. Por fim, o *f1-score* é calculado pela média harmônica da precisão do modelo (fração de acertos dentre as predições positivas) e a revocação do modelo (fração de acertos dentre a classe positiva) para avaliá-lo em casos com desbalanceamento de classes, ou seja, porcentagens baixas indicam que o modelo acerta simplesmente porque está enviesado pela classe mais representativa.

Observa-se na Tabela 2 que ambos os classificadores têm desempenho alto, com *Random Forest* ligeiramente superior a *SVM*. Especificamente, as suas porcentagens de acurácia e *auc-roc* estão acima de 81% indicando que ambos os modelos são hábeis para distinguir classes corretamente para a maioria dos objetos. Adicionalmente, a métrica *f1-score* tem valores acima de 70%, indicando que o desbalanceamento de objetos nas classes Quente e Morna, que é natural ao longo do tempo em serviços de armazenamento, não interfere nas predições de objetos na classe minoritária (Quente). O PoP-1 apresenta o melhor desempenho para ambos os classificadores, como pode ser observado na Tabela 2, devido o maior número de objetos o que favorece o treino do modelo.

Quanto à economia em custos de armazenamento, no traço PoP-2, alcançamos os maiores ganhos de economia com *r_{cs}* acima de 14%, ao passo que no traço PoP-1, o *r_{cs}* alcança 3,07%. Essa diferença ocorre devido uma característica do PoP-2, que é o menor tempo de vida para seus objetos (Seção 4.1), favorecendo a redução de custos. Adicionalmente, os modelos obtiveram a revocação mais expressiva comparada à precisão no PoP-2, o que leva menos falsos negativos.¹¹ Reduzir esse tipo de erro é importante pois ele tem impacto relevante no custo como mostra a Figura 2(a), i.e., operações e requisições a um objeto na classe Morna equivocadamente, pode tornar o custo final do objeto maior que o custo de armazená-lo na classe Quente. Além disso, o custo desse erro aumenta proporcionalmente ao volume do objeto. Para lidar com essa questão, balanceamos igualmente as classes nos treinos, o que contribuiu para aumentar acertos da classe minoritária (Quente), reduzindo falsos negativos como consequência. Ajustes nas configurações por classificador ainda podem ser utilizados para aumentar a economia de custos, e deixamos essa experimentação para trabalhos futuros.

¹⁰Adicionalmente foram avaliados Árvore de Decisão e Regressão Logística.

¹¹Prever um objeto inadequadamente para a classe Morna, quando ele ainda deveria estar na Quente.

5. Conclusões e Trabalhos Futuros

Nesse artigo, propomos um arcabouço para predição de classes de objetos em um serviço virtual de armazenamento em nuvem, visando redução de custos de armazenamento. Avaliamos esse arcabouço utilizando traços reais de acessos de usuários coletados do Dropbox. Os resultados mostraram que há um potencial para economia com redução de custos de armazenamento de até 25%, ou seja, aumento de lucratividade do serviço sem riscos de perda de qualidade para o usuário, ao explorar as classes Quente e Morna. Em cenários realistas, obter economia via modelos de predição, requer um arcabouço para gerenciamento automático dos objetos armazenados, como apresentamos nesse trabalho. Os modelos que avaliamos alcançaram acurácia, precisão e revocação superiores a 80% e uma economia de até 14% nos custos de armazenamento.

Trabalhos futuros incluem a avaliação do arcabouço proposto com três classes de objetos, acrescentando a classe Fria, e mecanismos de incentivos para compensar a perda na qualidade de serviço para o usuário com a classe Fria. Pretendemos também aprimorar o desempenho dos modelos de predição com configurações otimizadas por classificador para reduzir o custo de armazenamento.

Referências

- Bocchi, E., Drago, I., and Mellia, M. (2015). Personal Cloud Storage: Usage, Performance and Impact of Terminals. In *Proc. of the IEEE CloudNet*.
- Cisco (2019). Cisco Global Cloud Index: Forecast and Methodology, 2016–2021 White Paper. Disponível em <https://www.cisco.com> - Document ID 1513879861264127.
- Gonçalves, G. et al. (2016). The impact of content sharing on cloud storage bandwidth consumption. *IEEE Internet Computing*, 20(4):26–35.
- Gracia-Tinedo, R., García-López, P., Gómez, A., and Illana, A. (2016). Understanding data sharing in private personal clouds. In *Proc. of the IEEE CLOUD*.
- Hsu, Y., Irie, R., Murata, S., and Matsuoka, M. (2018). A novel automated cloud storage tiering system through hot-cold data classification. In *Proc. of the IEEE CLOUD*.
- Irie, R., Murata, S., Hsu, Y., and Matsuoka, M. (2018). A novel automated tiered storage architecture for achieving both cost saving and qoe. In *Proc. of the IEEE SC2*.
- Josep, A. D., Katz, R., Konwinski, A., Gunho, L., Patterson, D., and Rabkin, A. (2010). A view of cloud computing. *Communications of the ACM*, 53(4).
- Kaushik, R. T. and Bhandarkar, M. (2010). Greenhdfs: towards an energy-conserving, storage-efficient, hybrid hadoop compute cluster. In *Proc. of the USENIX*.
- Liu, M., Pan, L., and Liu, S. (2019). To transfer or not: An online cost optimization algorithm for using two-tier storage-as-a-service clouds. *IEEE Access*, 7:94263–94275.
- Mesnier, M., Ganger, G. R., and Riedel, E. (2003). Object-based storage. *IEEE Communications Magazine*, 41(8):84–90.
- Muralidhar, S. et al. (2014). f4: Facebook’s warm blob storage system. In *Proc. of the OSDI*.
- Ribeiro, S. O. et al. (2019). Análise de um serviço virtual de armazenamento em nuvem ciente de padrões de acesso por usuários. In *Proc. of the ERCEMAPI*.