

Diagnóstico de QoS em Redes a partir de Análise conjunta de Sobrevivência e Mudanças Estatísticas em Séries Temporais de Desempenho

Ian José Agra Gomes^{1,2}, Gabriel Bugginga¹, Edmundo de Souza e Silva¹,
Rosa Maria Meri Leão¹, Gaspare Bruno³

¹Programa de Engenharia de Sistemas e Computação
Universidade Federal do Rio de Janeiro (UFRJ)

²Centro de Análises de Sistemas Navais (CASNAV) – Marinha do Brasil

³Anlix – Brasil

{ian,buginga,edmundo,rosam,gaspares}@land.ufrj.br

Abstract. *This work presents a novel methodology for assessing the quality of service in computer networks. It combines time-series change-point detection techniques with clustering based on survival analysis. Motivated by a partnership between M-Lab and RNP, the methodology was applied to real data collected from an ISP using the NDT protocol. It allowed the identification of clients and servers with poorer performance than others. To facilitate the interpretation of the results by non-specialized teams, we used a large language model, which produces understandable reports from the methodology results.*

Resumo. *Este trabalho apresenta uma metodologia inovadora para avaliar a qualidade do serviço em redes de computadores, combinando técnicas de detecção de pontos de mudança em séries temporais com clusterização baseada em análise de sobrevivência. Motivado por uma parceria entre o M-Lab e a RNP, o método foi aplicado a dados reais, coletados de um ISP utilizando o protocolo NDT, permitindo identificar clientes e servidores com desempenho inferior aos demais. Para facilitar a interpretação dos resultados por equipes não especializadas, usamos um modelo de linguagem de grande porte (LLM), que produz relatórios compreensíveis a partir dos resultados da metodologia.*

1. Introdução

Modelos de confiabilidade e disponibilidade têm sido, há décadas, ferramentas essenciais para analisar vários aspectos de sistemas de computação [de Souza e Silva and Gail 1989]. Em Redes de Computadores, essas ferramentas também têm sido empregadas para avaliar a qualidade do serviço entregue aos usuários. Uma questão fundamental e não trivial em determinadas aplicações é a identificação de eventos que impactam negativamente a qualidade da rede, uma vez que as séries temporais de métricas de desempenho são geralmente não-estacionárias. Essa característica dificulta a elaboração de modelos preditivos e compromete a validade temporal dos rótulos de eventos, que, além de frequentemente escassos, tornam-se obsoletos com o tempo. Em geral, esse problema é conhecido como “detecção de ponto de mudança”, e tem encontrado aplicações em Redes, incluindo avaliação da qualidade do serviço [de Almeida et al. 2024].

A frequência de alterações em métricas de desempenho pode servir como indicador da qualidade da experiência do usuário de rede. Considere um determinado usuário e medições coletadas deste cliente para um dado servidor. Intuitivamente, se uma determinada métrica de desempenho, como a vazão ou latência medida de um usuário para um determinado servidor, tem suas estatísticas alteradas frequentemente, então este usuário provavelmente está experimentando um problema com a conexão a esse servidor.

Um desafio significativo surge quando precisamos analisar coletivamente dados provenientes de múltiplos clientes e servidores, considerando simultaneamente diferentes métricas de desempenho. Esse cenário ocorre, por exemplo, na análise de dados coletados com o protocolo NDT [M-Lab 2025] de diversos clientes para múltiplos servidores. Objetivamos responder questões fundamentais como: “Quais clientes apresentam um desempenho geral relativamente ruim?”; “Quais servidores possivelmente fornecem os piores resultados em suas medições?”; e “Quais métricas exercem maior impacto no desempenho global?”. Naturalmente, essas são perguntas muito amplas e difíceis de responder quando aplicadas a grandes volumes de dados.

Iremos explorar a intuição de que mudanças frequentes em métricas importantes de desempenho aliadas a técnicas de clusterização não-supervisionada são úteis para responder as questões apresentadas. Propomos uma metodologia inovadora, fundamentada em nossos trabalhos anteriores sobre análise de sobrevivência e detecção de pontos de mudança em séries temporais. Os pontos de mudança servem como delimitadores, segmentando as séries temporais em intervalos com características estatísticas distintas de seus vizinhos. Em seguida, aplicamos análise de sobrevivência a esses intervalos, agrupando-os em *clusters* que geram diferentes curvas de sobrevivência, as quais consideram diferentes métricas de interesse conjuntamente. A partir das curvas de sobrevivência resultantes da clusterização, mostramos como é possível fornecer respostas fundamentadas na metodologia.

Para tornar a avaliação acessível a especialistas sem base estatística, empregamos um modelo de linguagem generativo de grande porte (*LLM*) que, alimentado com as estatísticas e diretrizes específicas, produz análises técnicas compreensíveis para equipes não especializadas. Nossa contribuição é resumida a seguir:

Metodologia: desenvolvemos uma abordagem analítica que integra: (a) identificação não-supervisionada de pontos de mudança estatística em séries temporais com (b) clusterização de intervalos de mudança de diferentes séries em curvas de sobrevivência distintas.

Dataset: aplicamos a metodologia a um conjunto de dados reais coletados em um provedor parceiro, utilizando o protocolo NDT e servidores da Rede Nacional de Ensino e Pesquisa (RNP) e M-Lab.

Resultados: mostramos o potencial da análise para identificar clientes e/ou servidores com pior desempenho, traduzindo essas informações para um formato acessível a não especialistas.

2. Trabalhos Relacionados

A literatura sobre avaliação da Qualidade do Serviço (*Quality of Service* - QoS) em redes é extensa, incluindo diversos trabalhos que utilizam dados coletados por meio do protocolo NDT. No entanto, até onde sabemos, não há propostas que combinem *detecção de pontos*

de mudança em séries temporais com *análise de sobrevivência* aplicadas a métricas como latência e vazão para identificar, de forma não supervisionada, padrões recorrentes de degradação de desempenho em larga escala. A seguir, destacamos alguns trabalhos recentes por serem relevantes e complementares ao nosso trabalho.

Sharma et al. [Sharma et al. 2024a] investigam gargalos em redes de acesso residenciais, diferenciando as taxas de transferência entre cliente e roteador (Wi-Fi) e entre roteador e Internet (ISP). Com base em dados reais, os autores mostram que o Wi-Fi doméstico é frequentemente o fator limitante, especialmente entre usuários com planos de alta velocidade. Enquanto sua abordagem oferece uma análise micro do ambiente doméstico, nosso trabalho atua no nível macro da rede, analisando dados NDT de múltiplos clientes a servidores para detectar padrões de instabilidade ao longo do tempo.

Pi, Jamin e Wei [Pi et al. 2024] introduzem a ferramenta CongI, que combina detecção de mudanças na latência com classificadores SVM (*Support Vector Machine*) para identificar desequilíbrios de congestionamento em caminhos de rede balanceados. A proposta permite distinguir caminhos congestionados de alternativas subutilizadas em tempo quase real, com impactos significativos em métricas como vazão e perda de pacotes. Apesar do foco em rotas *anycast* e balanceamento de carga, assim como o nosso, utiliza séries temporais de latência para detectar degradações não triviais de desempenho. Enquanto CongI avalia problemas em caminhos de rede, nossa metodologia avalia a QoS a partir de medições realizadas entre pares cliente-servidor.

Sharma et al. [Sharma et al. 2024b] propõem uma abordagem baseada em clusterização temporal de latência aplicada a pequenas regiões geográficas, com o objetivo de agrupar áreas com padrões de desempenho similares, utilizando o 95^o percentil da latência como indicador de estabilidade. Embora compartilhem o objetivo de identificar padrões estruturais de degradação na rede, nossa abordagem foca na instabilidade observada em séries temporais individuais de clientes e servidores. As duas propostas se complementam ao oferecer visões espacial e temporal sobre a variabilidade da QoS.

3. Breve introdução às técnicas utilizadas

Neste trabalho, apresentamos uma metodologia capaz de identificar clientes que potencialmente experimentem algum problema de desempenho, utilizando análise não-supervisionada de séries temporais. Esta análise fundamenta-se em dois algoritmos: um para detecção de pontos de mudança em séries temporais [de Almeida et al. 2024] e outro que emprega conceitos de análise de sobrevivência para agrupar intervalos com características similares [Bugginga and de Souza e Silva 2024]. A seguir, descrevemos os aspectos fundamentais destes algoritmos.

O primeiro algoritmo, denominado *Voting Windows Change-point Detection* (VWCD) [de Almeida et al. 2024], foi concebido para detectar mudanças estatísticas em séries temporais em aplicações online, combinando eficiência computacional com simplicidade de implementação. O método utiliza janelas deslizantes que “votam” em um potencial ponto de mudança dentro de seu intervalo de observação. Cada ponto identificado como início de uma mudança significativa na série é acompanhado por um indicador numérico que expressa o grau de confiança dessa classificação, baseado na quantidade e no peso dos votos recebidos, o que torna os resultados facilmente interpretáveis.

O segundo algoritmo, *SurvMixClust*, é de clusterização e tem o seguinte obje-

tivo: dado um conjunto de intervalos e um número K de clusters, determinar as curvas de sobrevivência que melhor representam os intervalos associados a cada uma delas. Esse problema, tradicionalmente aplicado em análise de sobrevivência para estudos clínicos, permite identificar grupos de indivíduos com características e curvas de sobrevivência similares [de Souza e Silva et al. 2022]. Na área médica, os intervalos representam o tempo de vida dos indivíduos ou o período até o término do estudo (neste último, o intervalo é dito “censurado”). O algoritmo de clusterização proposto em [Buginga and de Souza e Silva 2024] apresenta características importantes, como a capacidade de aprender a função de sobrevivência de cada cluster, e o fato de ser não-paramétrico. O modelo é baseado em uma mistura de curvas de sobrevivência [Buginga and de Souza e Silva 2024]:

$$S_i(t|\mathbf{x}_i) = \sum_{k=1}^K \tau_k(\mathbf{x}_i) S_k(t|\theta_k). \quad (1)$$

onde, $S_i(t|\mathbf{x}_i)$ representa a função de sobrevivência em função do vetor de *features* \mathbf{x}_i para o indivíduo i , $S_k(t|\theta_k)$ é a função de sobrevivência (não-paramétrica) associada ao cluster k , e $\tau_k(\mathbf{x}_i)$ corresponde à regressão logística multinomial que modela as proporções da mistura de curvas de sobrevivência. Na regressão logística binária, a probabilidade condicional $p(y = 1 | x; \theta)$ é modelada por meio da função sigmoid aplicada a uma combinação linear dos valores das componentes do vetor de *features* \mathbf{x} e dos parâmetros θ : $p(y = 1 | x; \beta) = \sigma(\beta_0 + \beta_1 x_1 + \dots + \beta_D x_D)$, onde $\sigma(z)$ é a função sigmoid: $\sigma(z) = \frac{1}{1+e^{-z}}$, β_0 é o termo de viés, e β_1, \dots, β_D são os pesos associados às *features* x_1, \dots, x_D .

A aplicação deste algoritmo para análise de redes foi inicialmente proposta em [Buginga and de Souza e Silva 2024]. Neste trabalho, aplicamos essas ideias a dados recentes coletados em um ISP, e integrando-as com o VWCD. O algoritmo fornece uma tabela com os coeficientes β_i de cada *feature* x_i , incluindo variáveis discretas (no caso, clientes e servidores) e contínuas (estatísticas de vazão e RTT). Em nossa análise, esses coeficientes encontram-se listados na Tabela 1 para todas as *features* que usamos. A partir desses coeficientes, podemos inferir a qualidade, conforme será descrito na Seção 6.

4. Metodologia

Conforme introduzido na Seção 1, nossa metodologia é aplicável à análise de um conjunto grande de séries temporais. Ela tem como premissa a intuição de que mudanças estatísticas frequentes impactam negativamente a qualidade dos serviços oferecidos aos usuários, uma vez que alta variabilidade é intrinsecamente indesejável para métricas de desempenho. As séries temporais que usamos foram coletadas na rede de um ISP parceiro. As medições foram obtidas pelo protocolo NDT a partir de dispositivos Raspberry Pi distribuídos em pontos distintos da rede do provedor, conforme detalhado na Seção 5.

No primeiro passo da metodologia são identificados instantes de mudanças estatísticas de uma métrica de desempenho escolhida. Esses instantes servirão de base para a análise de sobrevivência que realizaremos posteriormente. Para cada par cliente-servidor, usamos o método proposto em [de Almeida et al. 2024] e a métrica “vazão de download” para identificar mudanças estatísticas. Entretanto, outras métricas podem ser consideradas, como a latência e a taxa de perda, por exemplo. Além disso, nosso método de

detecção de pontos de mudança permite considerar mudanças estatísticas analisando várias métricas simultaneamente, mas escolhemos apenas uma para facilitar a descrição dos resultados. O método também fornece informações adicionais que não foram exploradas neste trabalho.

A partir dos instantes de mudanças identificados, dividimos cada série em intervalos de tempo entre essas mudanças. Uma consideração importante é que nem todos os intervalos começam ou terminam em um ponto de mudança estatística. Precisamos levar em conta intervalos que terminam (ou têm início) no instante final (inicial) da coleta de dados, e aqueles que iniciam ou terminam devido a interrupções de medições. Esses são considerados como *intervalos censurados* [Buginga and de Souza e Silva 2024], e são marcados como tal e tratados automaticamente pelo algoritmo de clusterização.

Esse processo resulta em um conjunto de intervalos derivados de **todas** as séries temporais da métrica usada como base (no caso a vazão de download). Cada intervalo é associado a múltiplas *features*: a ID do cliente de onde partiu a medição relativa ao intervalo; a ID do servidor para onde foi feita a medição, um indicador de intervalo censurado ou não e métricas adicionais. Para este estudo, selecionamos oito *features* extras: vazões médias de upload e download no intervalo, RTT médio de download e upload durante o intervalo, e os respectivos desvios padrão. Outras métricas podem ser consideradas, mas escolhemos essas por serem comumente coletadas em redes de dados. Assim, cada intervalo associa-se a duas *features* discretas, oito *features* contínuas, um indicador de *censura*, e o tamanho do intervalo.

A seguir, utilizamos **todos** os intervalos de todas as séries como entrada do algoritmo de clusterização dos intervalos em K grupos, onde K é um parâmetro do modelo. Apenas para simplificar a exposição, neste trabalho escolhemos $K = 2$. Utilizamos o algoritmo recentemente proposto em [Buginga and de Souza e Silva 2024]. Esse algoritmo segmenta os intervalos em K conjuntos, agrupando aqueles que apresentam curvas de “sobrevivência” similares. É crucial ressaltar que a clusterização considera não apenas a duração dos intervalos, mas também as *features* associadas.

A Figura 2 ilustra as curvas de sobrevivência resultantes do exemplo a ser discutido. Como resultado, cada intervalo é associado a uma curva específica (vermelha ou azul). Conforme será detalhado na Seção 6, os intervalos marcados como vermelho são relativamente mais curtos que os de cor azul, além de compartilharem outras características similares dentro do mesmo cluster, como médias e desvios padrão de vazão e RTT.

O algoritmo de clusterização fornece os coeficientes do modelo obtido, conforme a Seção 3. Em nosso exemplo, esses coeficientes encontram-se listados na Tabela 1 para todas as *features*. Estes coeficientes são fundamentais para nossa análise, permitindo responder às questões propostas na Seção 1. Adicionalmente, o modelo permite determinar: (a) a probabilidade de um intervalo pertencer a um cluster específico; (b) a função de sobrevivência para novos intervalos caracterizados pelas *features* consideradas. Neste trabalho, somente utilizamos os coeficientes do modelo de clusterização. Estes são facilmente interpretáveis por especialistas ou, conforme mostraremos, podem ser processados por um LLM para emitir o resultado final a não-especialistas. A interpretação destes parâmetros para inferência de qualidade será apresentada na Seção 6.

A Figura 1 sintetiza o fluxo dos dados neste estudo.

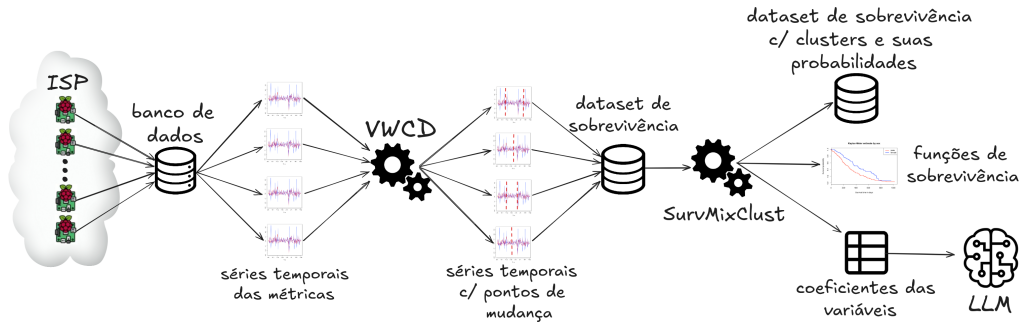


Figura 1. Fluxo de coleta e processamento dos dados.

5. Dataset

Coletamos nossos dados usando 16 dispositivos Raspberry Pi configurados para disparar um teste NDT a cada 30 minutos, ao longo de três meses, de maio a julho de 2024. O teste NDT, integrante do conjunto de ferramentas de medição do M-Lab, fornece várias métricas, incluindo vazão, latência e perda de pacotes, que influenciam a QoS oferecida pelo ISP [Streit et al. 2021]. Uma parceria recente M-Lab/RNP [RNP 2023] permitiu o uso da infraestrutura distribuída de servidores da RNP para a realização dos testes NDT, possibilitando a coleta de métricas mais representativas da experiência real do usuário por esses servidores estarem localizados em pontos estratégicos de troca de tráfego.

Nossa abordagem de medição difere da simples utilização de conjuntos de dados provenientes de testes NDT iniciados por usuários. Enquanto esses testes podem refletir na maioria situações em que os usuários percebem problemas de conexão e decidem avaliar o desempenho da sua rede, nossa coleta busca capturar um panorama abrangente e representativo das condições reais da rede. Ao realizar medições de forma sistemática e independente da percepção do usuário, reduzimos o viés introduzido pela tendência dos clientes de realizar testes apenas em momentos de degradação de QoS [Gill et al. 2022].

Conforme descrito na Seção 4, utilizamos o método VWCD para rotular as séries com os pontos de mudança detectados, e armazenamos também outras informações fornecidas pelo VWCD, além das estatísticas de cada métrica: média e desvio padrão da vazão e latência de download e upload.

Os pontos de mudança identificados nas séries temporais de vazão de download foram utilizados como eventos de interesse para a construção de um conjunto de dados de sobrevivência. Nesse *dataset*, cada entrada representa um intervalo temporal delimitado por pontos de mudança consecutivos. Os atributos associados a cada intervalo incluem: identificadores do cliente e do servidor (codificados em variáveis binárias por meio de *one-hot encoding*), duração do intervalo em dias, as estatísticas média e desvio padrão de cada métrica, e uma variável binária indicadora de ocorrência do evento ou de intervalo *censurado*.

6. Resultados

Utilizando os pontos de mudança na série temporal da vazão de download como eventos de interesse, construímos um conjunto de dados de sobrevivência para treinar o modelo *SurvMixClust* [Buginga 2023]. O algoritmo identificou dois clusters com características de sobrevivência notadamente distintas. Na Figura 2, o Cluster 0, em vermelho, exibe um

declínio mais acentuado na probabilidade de sobrevivência ao longo do tempo, indicando maior frequência de mudanças estatísticas nas séries correspondentes. Em contraste, o Cluster 1 contém intervalos mais longos entre eventos de mudanças estatísticas. Relembramos que todas as *features* associadas aos intervalos são consideradas na clusterização, e não apenas o tamanho do intervalo.

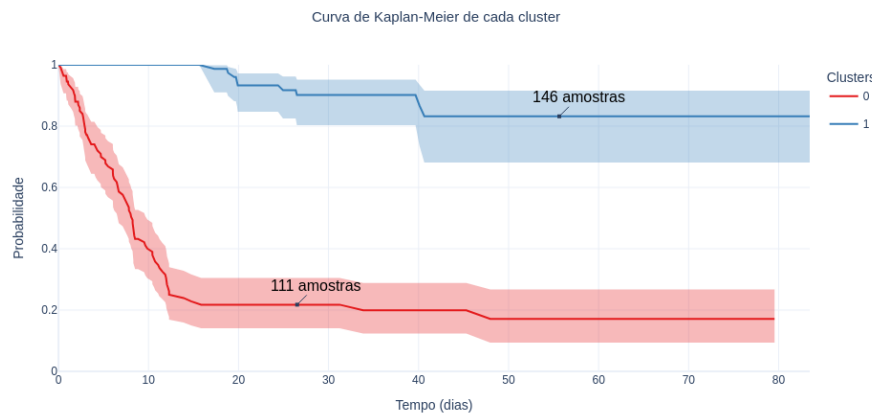


Figura 2. Funções de sobrevivência dos clusters (algoritmo *SurvMixClust*).

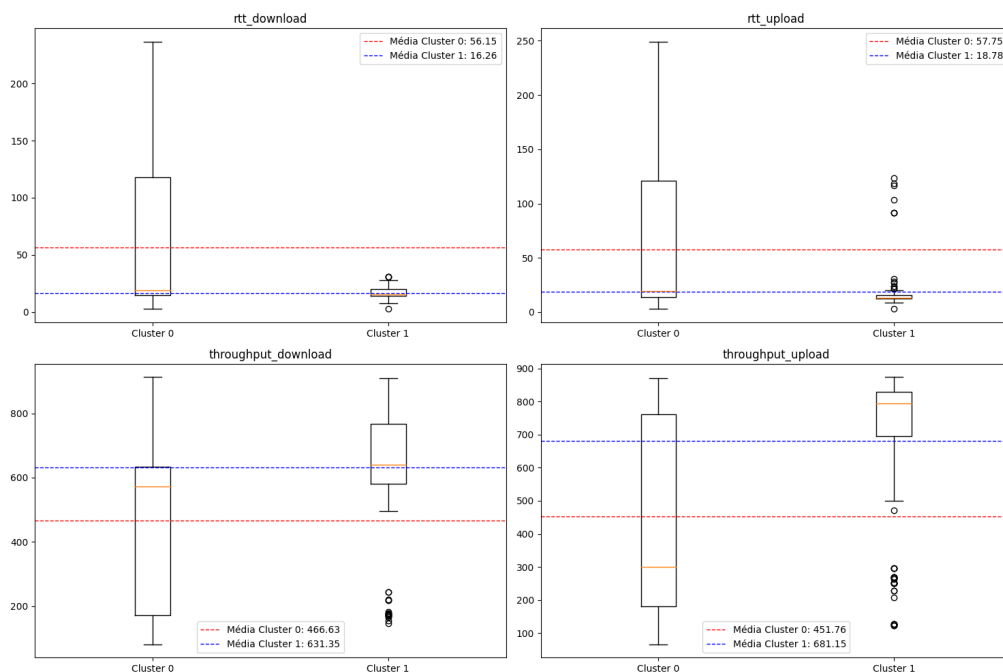


Figura 3. Boxplot das *features* analisadas para cada cluster.

Na Figura 3, a análise de *boxplots* confirma que as medições realizadas nos intervalos pertencentes ao Cluster 1 estão associadas a um desempenho médio superior: menor RTT médio, maior vazão média e menor variabilidade para todas essas métricas. Este cluster agrega intervalos mais longos entre eventos de mudanças estatísticas, sugerindo maior estabilidade e melhor qualidade da rede. As medições realizadas nos intervalos

pertencentes ao Cluster 0, por sua vez, resultam em RTT médio mais elevado, menor vazão média e variabilidade significativamente maior, portanto piores valores das métricas de desempenho.

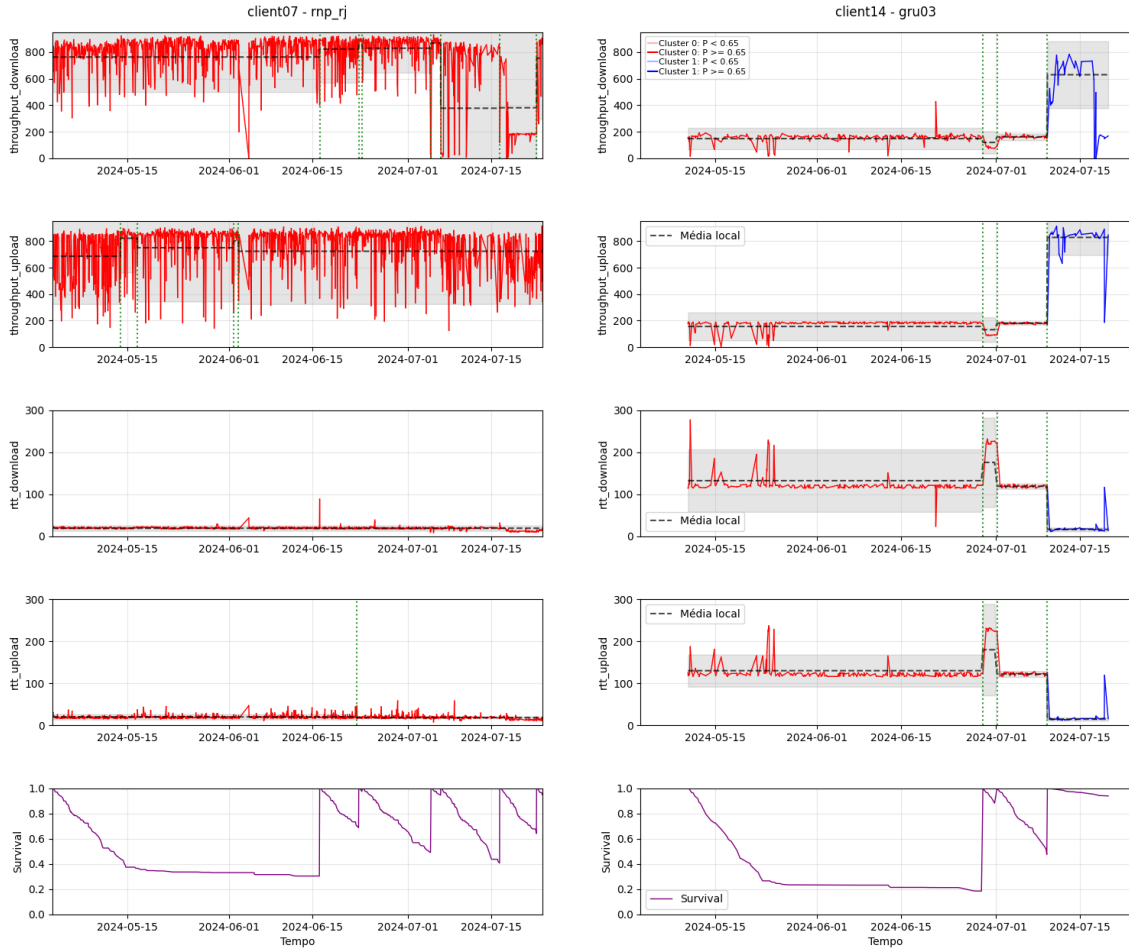


Figura 4. Gráficos das séries temporais das métricas de rede e da função de sobrevivência, para dois pares cliente-servidor.

O modelo permite calcular a probabilidade P de um intervalo de uma série temporal de um par cliente-servidor pertencer a um cluster, usando a Equação 2.

$$P(\text{Cluster 1}) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 X_1 + \beta_2 X_2 + \dots + \beta_k X_k)}}. \quad (2)$$

onde: $\beta_0, \beta_1, \beta_2, \dots, \beta_k$ são os coeficientes da regressão; X_1, X_2, \dots, X_k são os valores das *features* (e.g., média da vazão, média do RTT).

Selecionamos dois pares cliente-servidor para ilustrar a relação entre as séries temporais de métricas do cliente e a classificação de intervalos das séries em cada um dos clusters. A Figura 4 apresenta gráficos das métricas, e a cor identifica o cluster e a respectiva probabilidade (P). Além disso na figura estão indicados os instantes de mudança estatística que foram detectados (linhas verticais tracejadas de cor verde). Os gráficos na parte inferior representam a função de sobrevivência. A classificação dos intervalos é resultante da metodologia não supervisionada.

Foram identificados vários pontos de mudança na série “vazão de download” do par client07-rnp_rj. Esta característica, somada à alta variabilidade das vazões de download e upload, resultaram na classificação de todos os intervalos no Cluster 0 (vermelho).

O par client14-gru03 apresenta comportamento diferente do par client07-rnp_rj: inicialmente, os intervalos da série temporal “vazão de download” são classificados no Cluster 0. Apesar da pouca variabilidade, a classificação é provavelmente resultante da baixa vazão (inferior a 200 Mbps) e elevado RTT (em torno de 130 ms), o que mostra a relevância das outras *features* dos intervalos para a classificação, além do tamanho do intervalo. Os intervalos finais da série são associados ao Cluster 1, e é possível observar uma melhora significativa dos valores das métricas de desempenho no período azul.

Tanto os resultados dos boxplots quanto as análises de dois pares cliente-servidor reforçam a hipótese de que variações frequentes em uma das métricas de desempenho medida estão associadas a uma piora em outras métricas associadas ao mesmo cliente.

6.1. Análise dos resultados da regressão logística

Tabela 1. Coeficientes da regressão logística do modelo *SurvMixClust* e suas razões de chances (odds ratios) associadas a cada *feature*.

Categoria	Feature	Coeficiente	Odds Ratio	Categoria	Feature	Coeficiente	Odds Ratio
Clientes	client01	0.7487	2.11	Servidores	gru03	-0.4197	0.66
	client02	-1.9979	0.14		gru05	1.5070	4.51
	client03	0.6170	1.85		gru06	1.8974	6.67
	client04	0.2313	1.26		rnp_rj	-1.7618	0.17
	client05	1.3289	3.78		rnp_sp	-1.2229	0.29
	client06	-1.0035	0.37	Média	Vazão down	-0.0150	0.99
	client07	-1.5380	0.21		Vazão up	0.6859	1.99
	client08	0.3086	1.36		RTT down	-1.5921	0.20
	client09	0.0045	1.00		RTT up	-0.6199	0.54
	client10	0.0476	1.05	Desv. Padrão	Vazão down	-1.0413	0.35
	client11	0.7841	2.19		Vazão up	0.1860	1.20
	client12	0.0890	1.09		RTT down	-0.3346	0.72
	client13	-0.9369	0.39		RTT up	0.9051	2.47
	client14	0.7389	2.09				
	client15	0.5426	1.72				
	client16	0.0349	1.04				

A Tabela 1 apresenta os coeficientes da regressão logística e as respectivas *Odds ratios* para cada *feature* do modelo. Os coeficientes e as *Odds ratios* permitem interpretar a influência de cada *feature* no modelo de sobrevivência, isto é, na função que descreve os intervalos entre eventos de mudanças estatísticas em métricas da rede.

As *Odds ratios* de uma *feature* são calculadas pela exponencial $e^{\text{coeficiente}}$, e quantificam o efeito de uma *feature* como um fator multiplicativo sobre as chances do evento ocorrer. Valores superiores a 1 indicam que a *feature* aumenta as chances de pertencer ao Cluster 1, valores inferiores a 1 aumentam as chances de pertencer ao Cluster 0 e valores iguais a 1 indicam que a *feature* não tem influência sobre as chances do evento.

Na análise dos clientes, destacam-se client01, client03, client05, client11, client14 e client15 como clientes que aumentam as chances de pertencimento ao Cluster 1. O client05, por exemplo, apresenta 3.78 vezes mais chances de gerar intervalos longos. Em contrapartida, os clientes com *Odds ratios* menores do que 1, como client02, client06, client07 e client13, aumentam as chances de pertencer ao Cluster 0, associado a intervalos curtos entre eventos de mudanças.

Entre os servidores, gru05 e gru06 são fortemente associados a intervalos longos, com *odds ratios* significativamente superiores a 1, enquanto rnp_rj e rnp_sp aumentam as chances de pertencer ao Cluster 0.

As métricas de rede mostram que maior vazão média e menor RTT médio associam-se ao Cluster 1, confirmando que alta vazão impacta positivamente a QoS, enquanto alto RTT a afeta negativamente. A variabilidade das métricas também influencia significativamente: maior desvio padrão tanto na vazão quanto no RTT de download aumenta a probabilidade de pertencer ao Cluster 0, evidenciando a importância da estabilidade das métricas no desempenho da rede.

A Tabela 2 ilustra os valores da média e do desvio padrão das métricas para dois servidores, calculados a partir de todas as amostras das respectivas séries temporais. O servidor gru06 apresenta o coeficiente da regressão logística positivo e maior que 1, indicando alta probabilidade de pertencer ao cluster 1 (melhor QoS), enquanto o servidor rnp_rj apresenta coeficiente negativo, sinalizando maior chance de pertencer ao cluster 0 (pior QoS). É possível supor que um algoritmo simples, baseado apenas nos valores da média e do desvio padrão das métricas, seria suficiente para inferir a QoS dos clientes/servidores. Entretanto, a média e o desvio padrão das métricas desses dois servidores são bastante semelhantes, conforme mostrado na Tabela 2. Portanto, a média isoladamente não captura, por exemplo, a variabilidade temporal das medidas, e então é inadequada como único critério para avaliar a QoS.

Tabela 2. Estatísticas para servidores gru06 e rnp_rj.

Server	Mean RTT Down	STD RTT Down	Mean Vazão Down	STD Vazão Down	Mean RTT Up	STD RTT Up	Mean Vazão UP	STD Vazão UP	Coef. Reg. Log.
gru06	16.90	10.11	708.07	169.63	21.86	31.91	684.58	256.49	1.89
rnp_rj	18.68	5.95	664.74	165.39	24.46	25.10	628.95	257.89	-1.76

Os resultados permitem aplicações práticas imediatas. A identificação de *features* críticas, com *odds ratios* significativamente menores que 1 (como observado em client02 e rnp_rj), indicam que medições entre pares cliente/servidor (por exemplo, do client02 para qualquer servidor, ou de qualquer cliente para rnp_rj) podem passar por caminhos com problemas de conexão. Da mesma forma, clientes e servidores com “bom *Odds-ratio*”, como gru06 e client05, indicam boa qualidade de conexão.

A fim de permitir a reprodução de nossos experimentos, disponibilizamos o código-fonte, os dados e outras instruções em <https://github.com/ianagra/ndt-analysis>.

6.2. Resultado da Análise usando LLM

Nesta seção mostramos brevemente como é possível utilizar os resultados da nossa metodologia de forma simples, sem que seja necessário nenhum conhecimento especializado sobre os algoritmos de análise, baseando-se na Tabela 1. Essa tabela foi passada em formato JSON para um LLM, assim como detalhes da metodologia. Esses detalhes indicam que a série temporal base para achar os instantes de mudanças estatísticas é a “vazão de download”; que são achados instantes de mudanças estatísticas de cada série; que a divisão em dois *clusters* é feita pelo tamanho dos intervalos, a ocorrência (ou não) de evento e as *features* associadas; que regressão logística foi usada para o cálculo das probabilidades de um intervalo pertencer a um dos *clusters*. O *prompt* é complexo (omitido por

limitações de espaço) e a interpretação dos resultados é deixada para o LLM. O resultado pode ser detalhado, para o especialista, e também resumido para o não-especialista.

Fizemos testes com o mesmo prompt para três LLMs: (a) o modelo da OpenAI *GPT-4o*; (b) o modelo *Phi-4* [Abdin and *et al.* 2024], um LLM de 14 bilhões de parâmetros, o que possibilitou sua execução local em uma GPU Nvidia A4000 com 16Gb de VRAM; (c) o modelo *DeepSeek-R1* de 32 bilhões de parâmetros disponibilizado recentemente e usado também localmente.

O resultado da OpenAI é apresentado abaixo e foi traduzido para o Português usando o mesmo modelo de linguagem. Fornecemos ainda ao LLM as três questões fundamentais mencionadas na Seção 1. O resultado da análise para o não-especialista foi:

*** **Análise de Desempenho de Clientes e Servidores** ***

Apresentamos os resultados de desempenho com base nas medições realizadas pelos clientes e servidores monitorados. A análise identifica os clientes e servidores com pior desempenho e as métricas de desempenho mais relevantes.

* **Quais clientes estão com desempenho relativamente ruim no geral?**

Os seguintes clientes apresentam os piores resultados de desempenho, indicando maiores problemas de qualidade de serviço:

- **Client02:** Este cliente apresenta o pior desempenho geral, com impacto negativo significativo nas medições.
- **Client07 e Client13:** Estes clientes também apresentam problemas significativos, contribuindo para resultados de desempenho insatisfatórios.
- **Client06:** Embora não seja o pior, este cliente também está associado a resultados abaixo do esperado.

Ação recomendada: Investigar as condições desses clientes, como configurações de rede, padrões de tráfego ou possíveis limitações de infraestrutura.

* **Quais servidores provavelmente fornecem os piores resultados de desempenho?**

Os servidores que apresentam os piores resultados de desempenho são:

- **rnp_rj:** Este é o servidor com os piores resultados de desempenho geral.
- **rnp_sp:** Também apresenta problemas relevantes, mas com impacto menor que o servidor *rnp_rj*.
- **gru03:** Este servidor apresenta resultados abaixo do esperado, embora seu impacto negativo seja mais moderado.

Ação recomendada: Revisar a configuração e o desempenho desses servidores, priorizando a otimização de *rnp_rj* e *rnp_sp*, que são os mais críticos.

* **Quais métricas de desempenho estão impactando mais o desempenho?**

Entre as métricas analisadas, as mais impactantes são:

- **RTT de Download:** Esta métrica tem o impacto negativo mais significativo. Valores altos de RTT de download indicam maior latência, o que prejudica a qualidade do serviço.
- **Desvio Padrão da Vazão de Download:** Alta variabilidade na vazão de download indica instabilidade, afetando negativamente a experiência dos usuários.
- **Vazão de Upload:** Um aumento na vazão de upload está associado a melhorias no desempenho, indicando que essa métrica contribui positivamente.

Ação recomendada: Trabalhar para reduzir a latência do RTT de download e a variabilidade da vazão de download, além de garantir boas taxas de vazão de upload.

Resumo e Próximos Passos

Os resultados destacam áreas críticas para melhorar a qualidade de serviço:

- **Foco nos clientes problemáticos:** Revisar as configurações e identificar gargalos de desempenho relacionados a *Client02*, *Client07* e *Client13*.
- **Otimização de servidores:** Priorizar intervenções nos servidores *rnp_rj* e *rnp_sp*.
- **Redução de latência e variabilidade:** Investir em melhorias na latência e estabilidade da vazão de download.

Essas ações ajudarão a elevar a qualidade do serviço oferecido aos clientes e a melhorar o desempenho geral da rede.

É interessante ressaltar que os resultados dos outros LLMs usados foram os mesmos, variando apenas a forma do texto. Isso indica que é possível usar, para essa tarefa, modelos menores e de código aberto, executados localmente, acoplados à nossa metodologia. Por limitações de espaço, não incluímos os outros resultados.

7. Conclusões

Neste trabalho, respondemos questões aparentemente simples apresentadas na Seção 1, mas que apresentam desafios significativos em sua resolução. Nossa contribuição principal reside na integração de duas técnicas estatísticas que desenvolvemos recentemente para produzir coeficientes de fácil interpretação. Mostramos que estes coeficientes permitem a formulação de respostas objetivas para problemas complexos. Adicionalmente, evidenciamos que, mesmo um LLM com arquitetura relativamente modesta, executado

localmente e alimentado apenas com os coeficientes obtidos e algumas instruções básicas da aplicação, pode auxiliar na elaboração de relatórios acessíveis para usuários sem conhecimento estatístico avançado. Por fim, os resultados apresentados neste trabalho basearam-se em dados reais coletados em um ISP parceiro e usando o protocolo NDT.

Referências

- Abdin, M. I. and *et al.* (2024). Phi-4 technical report. Technical Report MSR-TR-2024-57, Microsoft.
- Buginga, G. and de Souza e Silva, E. (2024). Clustering survival data using a mixture of non-parametric experts. *arXiv*.
- Buginga, G. C. (2023). *Algoritmos de Aprendizado de Máquina Aplicados a Dados Censurados para Previsão de Mortalidade de Pacientes com Doença Arterial Coronariana*. Dissertação de mestrado, COPPE - Universidade Federal do Rio de Janeiro.
- de Almeida, C. M., Leão, R. M. M., and de Souza e Silva, E. (2024). Inferindo pontos de mudança em séries temporais com dados não rotulados: um breve estudo usando dados do NDT. In *XLII SBRC*, pages 686–699.
- de Souza e Silva, C. G., Buginga, G. C., de Souza e Silva, E. A., Arena, R., Rouleau, C. R., Aggarwal, S., Wilton, S. B., Austford, L., Hauer, T., and Myers, J. (2022). Prediction of mortality in coronary artery disease: Role of machine learning and maximal exercise capacity. *Mayo Clinic Proceedings*, 97(8):1472–1482.
- de Souza e Silva, E. and Gail, H. R. (1989). Calculating availability and performability measures of repairable computer systems using randomization. *J. ACM*, 36(1):171–193.
- Gill, P., Diot, C., Ohlsen, L. Y., Mathis, M., and Soltesz, S. (2022). M-lab: user initiated internet data for the research community. *SIGCOMM Comput. Commun. Rev.*, 52(1):34–37.
- M-Lab (2025). Ndt (network diagnostic tool).
- Pi, Y., Jamin, S., and Wei, L. (2024). Measuring congestion-induced performance imbalance in internet load balancing at scale. In *Proceedings of the ACM SIGCOMM*.
- RNP (2023). RNP e M-Lab assinam MoU. <https://www.rnp.br/rnp-e-m-lab-assinam-mou-para-disponibilizacao-de-medies-de-desempenho-de-rede-para-fins-de-pesquisa>.
- Sharma, A., Sundaresan, S., and Madhyastha, H. V. (2024a). Measuring the prevalence of wifi bottlenecks in home access networks. In *Proceedings of the ACM Internet Measurement Conference (IMC)*.
- Sharma, T., Padaki, R., and Feamster, N. (2024b). Beyond data points: Regionalizing crowdsourced latency measurements. In *Proceedings of the ACM SIGMETRICS*. ACM.
- Streit, A. G., dos Santos, G. H. A., Leão, R. M. M., e Silva, E. d. S., Menasché, D., and Towsley, D. (2021). Network anomaly detection based on tensor decomposition. *Computer Networks*, 200:108503.