



UNIVERSIDADE FEDERAL DE PERNAMBUCO  
CENTRO DE INFORMÁTICA  
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIAS DA COMPUTAÇÃO

FELIPE AUGUSTO MARQUES DE ALCÂNTARA

**ESTRATÉGIAS DE TREINAMENTO EM BATCH PARA CLASSIFICADORES  
BINÁRIOS QUÂNTICOS BASEADOS EM GRADIENTE**

Recife

2025

FELIPE AUGUSTO MARQUES DE ALCÂNTARA

**ESTRATÉGIAS DE TREINAMENTO EM BATCH PARA CLASSIFICADORES  
BINÁRIOS QUÂNTICOS BASEADOS EM GRADIENTE**

Dissertação apresentada ao Programa de Pós-Graduação em Ciência da Computação da Universidade Federal de Pernambuco, como requisito parcial para obtenção do título de mestre em Ciência da Computação.

**Área de Concentração:** Inteligência Computacional

**Orientador:** Adenilton J. Silva

Recife

2025

.Catalogação de Publicação na Fonte. UFPE - Biblioteca Central

Alcântara, Felipe Augusto Marques de.

Estratégias de treinamento em batch para classificadores binários quânticos baseados em gradiente / Felipe Augusto Marques de Alcântara. - Recife, 2025.

60f.: il.

Disertação (Mestrado)- Universidade Federal de Pernambuco, Centro de Informática, Programa de Pós-Graduação em Ciência da Computação, 2025.

Orientação: Adenilton José da Silva.

1. Computação quântica; 2. Classificadores quânticos; 3. Aprendizagem em batch; 4. Algoritmos variacionais. I. Silva, Adenilton José da. II. Título.

UFPE-Biblioteca Central

**Felipe Augusto Marques de Alcântara**

**“Estratégias de Treinamento em Batch para Classificadores Binários  
Quânticos Baseados em Gradiente”**

Dissertação de mestrado apresentada ao Programa de Pós-Graduação em Ciência da Computação da Universidade Federal de Pernambuco, como requisito parcial para a obtenção do título de Mestre em Ciência da Computação. Área de Concentração: Inteligência Computacional

Aprovado em: 22/08/2025.

**BANCA EXAMINADORA**

---

Prof. Dr. Adenilton José da Silva  
Centro de Informática / UFPE  
(**orientador**)

---

Prof. Dr. Nadja Kolb Bernardes  
Departamento de Física / UFPE

---

Prof. Dr. Péricles Barbosa Cunha de Miranda  
Departamento de Computação / UFRPE



Dedico à criança indagadora e curiosa que inventivamente cria e se entrete.

## AGRADECIMENTOS

Todo trabalho acadêmico é fruto de muito esforço para formular e responder questionamentos que se situam nas fronteiras do conhecimento. Aqueles que se lançam nesse empreendimento o fazem de forma parecida aos primeiros navegantes que embarcaram em navios de madeira sem saber ao certo onde os ventos e as águas os levariam.

Assim como os primeiros navegantes, o pesquisador se subtrai a momentos de convivência com aqueles que lhe são mais caros para mergulhar na incerteza do desconhecido. O que fica para aqueles que o amam? ausência e compreensão.

Lidar com a ausência não é tarefa trivial. Compreender que o sonho e as ambições daqueles que amamos por vezes os levam para longe do nosso convívio é empresa difícil e deveras assustadora. Agradeço imensamente aos meus pais, Maria Sebastiana Marques de Lima Alcântara e José Lucinaldo de Alcântara Nascimento, e a minha irmã, Bárbara Gabrielle Marques de Alcântara, por terem parcimoniosamente convivido com meus momentos de ausência e reclusão. Agradeço, ainda, as ininterruptas insistências da minha irmã para que eu empunhasse novamente o lápis e ocupasse lugar nos bancos escolares com uma nova roupagem: a roupagem de quem produz conhecimento e não apenas consome o que está escrito nos livros.

Nenhuma nau é capaz de partir com um homem apenas. É necessário alguém com experiência, alguém que saiba manipular a bússola e o astrolábio. Alguém que saiba identificar a direção dos ventos e os indícios de intempéries. Acredito que o papel do orientador seja como o de capitão de um navio onde o traço unilateral e vertical de transmissão de ordens dá lugar a um espaço enriquecedor de troca de ideias e produção de conhecimento. Agradeço fortemente ao meu orientador, Adenilton José da Silva, por todas as trocas e direções que me engrandeceram e me inspiraram durante essa jornada. Agradeço, também, aos corajosos tripulantes do grupo de pesquisa de computação quântica. Em especial, Ismael, Jefferson, Israel, Thiago e Ana.

Por fim, externo meus sinceros agradecimentos ao Instituto de Ciência e Tecnologia Itaú pelo suporte financeiro. Sem o investimento em ciência, toda ideia ocupa lugar apenas nos livros e no saber axiomático.

"(...) o pensamento, no nível de sua existência, desde sua forma mais matinal, é, em si mesmo, uma ação - um ato perigoso." (FOUCAULT, 2002).

## RESUMO

O processamento da informação quântica inaugura uma nova forma de computação e uma nova forma de pensar técnicas de processamento de dados para aprendizagem de máquinas. Este estudo investiga o uso da superposição na construção de batches de dados. Trabalhos como (FARHI; NEVEN, 2018) e (SCHULD et al., 2020) introduziram a ideia mas até o presente momento não existe uma análise comparativa ou mesmo uma formalização conceitual sobre o tema. De forma a preencher essa lacuna, propõe-se um método comparativo e uma categorização das estratégias encontradas na literatura. Como contribuição adicional, apresenta-se uma nova estratégia em que o batch é composto por mais de um circuito com dados em superposição. As estratégias foram avaliadas de forma experimental em classificadores quânticos variacionais para diferentes configurações de hiperparâmetros do batch. Entre os resultados, verificou-se que estratégias com maior quantidade de padrões em superposição alcançaram valores médios superiores de acurácia no treino e no teste, além de maior capacidade de generalização.

**Palavras-chaves:** computação quântica; classificadores quânticos; aprendizagem em batch; algoritmos variacionais.

## ABSTRACT

Quantum information processing inaugurates a new form of computation and a new way of designing data processing techniques for machine learning. This study investigates the use of superposition in the construction of data batches. Works such as (FARHI; NEVEN, 2018) and (SCHULD et al., 2020) introduced the idea, but up to now there has been neither a comparative analysis nor a conceptual formalization of the subject. To fill this gap, a comparative method and a categorization of the strategies found in the literature are proposed. As an additional contribution, a new strategy is presented in which the batch is composed of more than one circuit with data in superposition. The strategies were experimentally evaluated in variational quantum classifiers under different batch hyperparameter configurations. Among the results, it was observed that strategies with a higher number of superposed patterns achieved superior average accuracy in both training and testing, as well as greater generalization capacity.

**Keywords:** quantum computing; quantum classifiers; batch learning; variational algorithms.

## LISTA DE FIGURAS

Figura 1 – Exemplo de circuito quântico . . . . .	22
Figura 2 – Exemplo de ansatz . . . . .	25
Figura 3 – Modelo de aprendizagem baseado em circuito quântico parametrizado . . .	29
Figura 4 – Exemplo de batch sequencial de tamanho 4 . . . . .	33
Figura 5 – Exemplo de batch sequencial de tamanho 4 . . . . .	34
Figura 6 – Batch Superposição - caso geral . . . . .	34
Figura 7 – Exemplo de batch duplo de tamanho 2 com superposições de tamanho 2 .	36
Figura 8 – Exemplo de entrada com 4 amostras carregadas em superposição . . . . .	38
Figura 9 – Exemplo de modelo com 3 qubits e 2 camadas . . . . .	39
Figura 10 – Exemplo de circuito com entrada, modelo e medição . . . . .	40
Figura 11 – Resultados IRIS para $n = 2$ . . . . .	47
Figura 12 – Resultados IRIS para $n = 4$ . . . . .	49
Figura 13 – Resultados IRIS para $n = 8$ . . . . .	50
Figura 14 – Resultados MNIST para $n = 2$ . . . . .	53
Figura 15 – Resultados MNIST para $n = 4$ . . . . .	55
Figura 16 – Resultados MNIST para $n = 8$ . . . . .	56

## LISTA DE CÓDIGOS

## LISTA DE QUADROS



## LISTA DE TABELAS

Tabela 1 – Operadores quânticos com representações matriciais e circuitais. . . . .	22
Tabela 2 – Síntese das expressões quantitativas de padrões e shots por circuito no batch	44
Tabela 3 – Hiperparâmetros para as estratégias sequencial e superposição . . . . .	45
Tabela 4 – Hiperparâmetros para a estratégia de batch duplo . . . . .	46
Tabela 5 – Melhores acurácias de treino e teste para $n = 2$ . . . . .	46
Tabela 6 – Melhores acurácias de treino e teste para $n = 4$ . . . . .	48
Tabela 7 – Melhores acurácias de treino e teste para $n = 8$ . . . . .	51
Tabela 8 – Melhores acurácias de treino e teste para $n = 2$ . . . . .	52
Tabela 9 – Melhores acurácias de treino e teste para $n = 4$ . . . . .	54
Tabela 10 – Melhores acurácias de treino e teste para $n = 8$ . . . . .	54

## LISTA DE SÍMBOLOS

$\nabla$	Operador gradiente
$\mathcal{C}$	Custo
$\mathcal{D}$	Conjunto de dados
$\mathcal{E}$	Operador de codificação em estado quântico
$\mathcal{M}$	Operador do modelo de aprendizagem
$\langle \cdot \rangle_{\theta}$	Valor esperado para o parâmetro $\theta$
$s$	Quantidade de shots
$n$	Quantidade de padrões no batch
$\mu$	Densidade de shots no batch
$b$	Quantidade de circuitos no batch

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO</b>	<b>16</b>
1.1	MOTIVAÇÃO	17
1.2	OBJETIVOS	19
<b>1.2.1</b>	<b>Objetivos Específicos</b>	<b>19</b>
<b>2</b>	<b>FUNDAMENTAÇÃO TEÓRICA</b>	<b>21</b>
2.1	COMPUTAÇÃO QUÂNTICA	21
<b>2.1.1</b>	<b>Evolução de Estados</b>	<b>21</b>
<b>2.1.2</b>	<b>Medida</b>	<b>22</b>
2.2	ALGORITMOS QUÂNTICOS VARIACIONAIS	23
<b>2.2.1</b>	<b>Função de Custo</b>	<b>24</b>
<b>2.2.2</b>	<b>Ansatz</b>	<b>24</b>
<b>2.2.3</b>	<b>Gradientes</b>	<b>25</b>
<b>2.2.4</b>	<b>Otimização</b>	<b>25</b>
2.3	APRENDIZAGEM DE MÁQUINAS	26
<b>2.3.1</b>	<b>Classificação Binária</b>	<b>26</b>
<b>2.3.2</b>	<b>Classificadores Quânticos</b>	<b>27</b>
2.3.2.1	<i>Codificação</i>	27
<b>2.3.2.1.1</b>	<b>Base</b>	<b>28</b>
<b>2.3.2.1.2</b>	<b>Amplitude</b>	<b>28</b>
<b>2.3.2.1.3</b>	<b>Ângulo</b>	<b>29</b>
2.3.2.2	<i>Processamento</i>	29
<b>2.3.2.2.1</b>	<b>Modelos Quânticos Determinísticos</b>	<b>29</b>
<b>2.3.2.2.2</b>	<b>Modelos Quânticos Probabilísticos</b>	<b>30</b>
2.3.2.3	<i>Regra de Decisão</i>	30
2.4	CONSIDERAÇÕES FINAIS	30
<b>3</b>	<b>BATCH QUÂNTICO</b>	<b>32</b>
3.1	BATCH CLÁSSICO	32
3.2	O EQUIVALENTE QUÂNTICO	33
<b>3.2.1</b>	<b>Batch Sequencial</b>	<b>33</b>
<b>3.2.2</b>	<b>Batch Superposição</b>	<b>33</b>

3.2.3	<b>Batch Duplo</b>	36
3.3	CONSIDERAÇÕES FINAIS	36
4	<b>METODOLOGIA</b>	38
4.1	PROCESSAMENTO QUÂNTICO	38
4.1.1	<b>Entrada</b>	38
4.1.2	<b>Modelo</b>	39
4.1.3	<b>Saída</b>	39
4.2	CONFIGURAÇÃO EXPERIMENTAL	40
4.2.1	<b>Bases de Dados</b>	40
4.2.1.1	<i>IRIS</i>	40
4.2.1.2	<i>MNIST</i>	41
4.2.2	<b>Simulador Quântico</b>	41
4.2.3	<b>Pré-Processamento dos Dados</b>	41
4.2.4	<b>Otimizador</b>	42
4.2.5	<b>Métricas de Avaliação</b>	42
5	<b>RESULTADOS</b>	45
5.1	IRIS	46
5.1.1	$n = 2$	46
5.1.2	$n = 4$	47
5.1.3	$n = 8$	48
5.1.4	<b>Síntese dos Resultados</b>	51
5.2	MNIST	51
5.2.1	$n = 2$	51
5.2.2	$n = 4$	52
5.2.3	$n = 8$	52
5.2.4	<b>Síntese dos Resultados</b>	54
6	<b>CONCLUSÃO</b>	57
	<b>REFERÊNCIAS</b>	58

## 1 INTRODUÇÃO

A constatação de que a mecânica quântica é capaz de produzir padrões contra intuitivos nos dados e com difícil reprodutibilidade em um computador clássico tem gerado a expectativa de possíveis ganhos em aplicações de computação quântica na área de aprendizagem de máquinas (BIAMONTE et al., 2017).

A computação quântica consiste em um modelo de processamento da informação baseado nas leis da mecânica quântica (NIELSEN; CHUANG, 2010). Algoritmos que fazem uso desse modelo buscam superar limitações no modelo clássico e obter algum tipo de aceleração ou melhoria de desempenho. Limitações como a fatoração de inteiros em tempo polinomial e a simulação de sistemas físicos de forma eficiente já têm soluções algorítmicas quânticas com vantagens teóricas conhecidas (MONTANARO, 2016).

A aprendizagem de máquinas é um ramo de inteligência artificial que habilita computadores à executar tarefas sem serem explicitamente programados. De forma geral, um modelo clássico de aprendizagem faz uso de um conjunto de dados para alimentar um algoritmo que iterativamente busca maximizar uma medida de desempenho. Robótica, veículos autônomos e reconhecimento de discurso são algumas aplicações populares de aprendizagem de máquinas presentes no dia-a-dia (ALZUBI; NAYYAR; KUMAR, 2018).

Um crescente número de estudos têm buscado explorar diferentes abordagens do que se entende como aprendizagem de máquinas quântica. Lamichhane e Rawat (2025) apresentam 4 principais abordagens que se distinguem na forma como os recursos quânticos são utilizados no contexto de aprendizagem de máquinas. Schuld, Sinayskiy e Petruccione (2015) discutem o uso da computação e dos algoritmos quânticos como ferramentas de suporte à aprendizagem mas apontam que a formalização conceitual de uma teoria quântica de aprendizagem permanece obscura.

Embora muito já se tenha explorado nas formas de concatenar as disciplinas de aprendizagem de máquinas e de computação quântica, o desenvolvimento do dispositivo físico para materializar essas ideias se encontra em seus estágios iniciais de desenvolvimento. Isso motivou direcionar esforços de pesquisa para compreender o que é possível ser feito com o atual hardware disponível gerando possivelmente evidências práticas de vantagens no domínio de processamento de informação quântica (BHARTI et al., 2022).

Uma das abordagens que se mostra bastante promissora para o dispositivo quântico exis-

tente na atualidade é a utilização de modelos de aprendizagem de máquina baseados em circuitos quânticos parametrizados. Essa abordagem guarda similaridades com modelos clássicos consagrados (redes neurais e métodos de kernel), justificando parte do sucesso em problemas de classificação, regressão e modelagem generativa (BENEDETTI et al., 2019). Para tarefas de aprendizagem supervisionada onde o objetivo é encontrar um mapeamento de um conjunto de entradas para um conjunto de saídas, os métodos de estimação de kernel quântico e algoritmos quânticos variacionais têm sido amplamente utilizados (JERBI et al., 2023).

A estimação de kernel quântico consiste basicamente em fazer um mapeamento da informação para um espaço de dimensão mais alta (espaço de Hilbert) que tenha uma fronteira de decisão linear e acessá-lo através de operadores de medição. Esse modelo de aprendizagem guarda uma relação muito próxima com os métodos de kernel em aprendizagem clássica, distinguindo-se dele apenas na forma como a saída é extraída do modelo (SCHULD, 2021).

A abordagem baseada em algoritmos variacionais faz uso de portas parametrizadas que são ajustadas a cada apresentação de um novo padrão. O ajuste é feito classicamente utilizando uma função de perda construída com base na saída do processador quântico (CEREZO et al., 2021). Classificadores (MAHESHWARI; SIERRA-SOSA; GARCIA-ZAPIRAIN, 2021), autoencoders (ROMERO; OLSON; ASPURU-GUZI, 2017) e redes neurais (CONG; CHOI; LUKIN, 2019) estão entre as aplicações de algoritmos variacionais no contexto de aprendizagem de máquinas.

Essas estratégias pensadas para o processador quântico atual buscam fazer uso de recursos quânticos, como superposição e emaranhamento, para obter algum tipo de vantagem comparativamente à forma clássica de processamento. O que torna ainda mais custosa a busca por bons algoritmos quânticos é o requisito de que eles já devem nascer com a promessa de resolver intratabilidades de algoritmos clássicos (NIELSEN; CHUANG, 2010).

## 1.1 MOTIVAÇÃO

O dispositivo quântico disponível não tem o potencial de implementar as principais soluções que evidenciam, de forma teórica, as vantagens da computação quântica. As limitações na quantidade e conectividade de qubits associadas aos erros de decoerência impõem restrições e disciplinam novas formas de uso (CEREZO et al., 2021).

Algoritmos variacionais são uma das principais abordagens para explorar evidências práticas das vantagens da computação quântica com o dispositivo atual. Esses algoritmos também são conhecidos como análogos quânticos das redes neurais e utilizam otimizadores clássicos como

parte das etapas de processamento (CEREZO et al., 2021).

No contexto de aprendizagem quântica variacional, otimizadores baseados em gradiente apresentam desempenho superior em relação à abordagem livre de gradiente (LOCKWOOD, 2022). O gradiente descendente, juntamente com suas versões estocástica e de mini-batch, estão entre as técnicas mais populares desse modelo de otimização (RUDER, 2016) e se diferenciam pela forma como os dados são apresentados.

Métodos de otimização estocástica em aprendizagem clássica consideram subconjuntos de tamanho fixo do conjunto de treinamento para atualizar iterativamente os pesos (GOODFELLOW et al., 2016). O uso mais eficiente da informação e passos iterativos de computação menos custosos são algumas das motivações a favor dos métodos estocásticos comparados ao método determinístico (BOTTOU; CURTIS; NOCEDAL, 2018).

Outra estratégia consiste em aproveitar o melhor que os métodos estocástico e determinístico tem a oferecer. Friedlander e Schmidt (2012) propõem um método de otimização híbrida determinístico-estocástica que inicia com gradiente incremental e paulatinamente aumenta a quantidade de amostras de forma a preservar um rápido progresso inicial sem sacrificar uma taxa de convergência atrativa.

O advento de um novo modelo de computação permite ampliar o repertório de estratégias de otimização em aprendizagem de máquinas supervisionada baseada em gradiente. Sweke et al. (2020) exploram novas formas de otimização estocástica em contextos onde processadores quântico e clássico trabalham de forma colaborativa em algoritmos variacionais. Kübler et al. (2020) propõem uma otimização adaptativa baseada na quantidade de medições para estimar observáveis em arquiteturas de processamento híbridas.

No contexto de aprendizagem de máquinas quântica, algumas técnicas baseadas em superposição de estados tem sido exploradas em conjunção com otimização estocástica baseada em gradiente. Farhi e Neven (2018) consideram um problema de classificação binária e codificam subconjuntos do conjunto de treino como a superposição de estados baseados em amostras pertencentes à uma mesma classe. Schuld et al. (2020) esboçam uma ideia de otimização com dados em superposição. Duan, Sun e Hsieh (2024) exploram o comportamento de otimização estocástica com um subconjunto de treino fixo e codificado em superposição.

Apesar dos trabalhos relacionados existentes na literatura introduzirem uma ideia a respeito de como recursos quânticos podem ser utilizados para apresentar padrões de forma alternativa, não existe uma formalização conceitual de batch que seja de uso comum entre eles e permita uma comunicação clara sobre o tema. Ademais, as formas alternativas de apresentação de

padrões ocupam uma importância secundária e ainda pouco avaliadas de forma analítica e experimental.

Esse trabalho explora estratégias de otimização estocástica em aprendizagem supervisionada com base em algoritmos variacionais. Serão apresentadas 3 técnicas de apresentação de padrões para passos iterativos de otimização. Em virtude da ausência de formas comparativas de estratégias de batch no domínio quântico, será definido um critério quantitativo de comparação para avaliar as estratégias apresentadas quanto à convergência e às acurácias de treino e teste.

## 1.2 OBJETIVOS

O objetivo geral desse trabalho é avaliar, de forma experimental, o uso de diferentes técnicas de apresentação de padrões em modelos de aprendizagem supervisionada baseados em gradiente.

### 1.2.1 Objetivos Específicos

1. Definir estratégias de apresentação de padrões;
2. Definir critério de valoração das estratégias apresentadas;
3. Avaliar a importância de hiperparâmetros relacionados à natureza quântica de processamento da informação no desempenho de modelos de aprendizagem supervisionada.

A formalização conceitual de estratégias de treinamento em batch com o uso de recursos quânticos e um critério para valorá-las estão entre as principais contribuições teóricas alcançadas no presente trabalho. Do ponto de vista prático, os resultados numéricos obtidos para dois conjuntos de dados evidenciam que, em arquiteturas executadas com baixo número de medições por circuito, um maior nível de superposição entre os padrões acelera a convergência e permite obter maiores valores médios de acurácias no treino e no teste.

Nos capítulos que seguem, serão apresentados fundamentos teóricos, conceitos autóctones e resultados experimentais. O capítulo 2 introduz as principais ideias sobre computação quântica e aprendizagem de máquinas utilizados nos capítulos seguintes. O capítulo 3 desenvolve o conceito das estratégias de batch sequencial, em superposição e duplo. O capítulo 4 especifica



o aparato experimental e o critério comparativo utilizado. O capítulo 5 apresenta e discute os resultados obtidos. Por fim, o capítulo 6 sintetiza as principais contribuições e sugere direções de trabalhos futuros.

## 2 FUNDAMENTAÇÃO TEÓRICA

### 2.1 COMPUTAÇÃO QUÂNTICA

A computação quântica é um campo de conhecimento que investiga o uso dos princípios da mecânica quântica para o processamento da informação. Diferentemente do modelo de computação tradicional que utiliza lógica booleana para processar informação determinística, a computação quântica utiliza um conjunto específico de propriedades para processar informação probabilística (NIELSEN; CHUANG, 2010).

A unidade básica de informação quântica é o bit quântico ou qubit. Essa unidade consiste em uma abstração matemática para sistemas físicos de 2 níveis, a exemplo do spin de um elétron ou da polarização de um fóton. O qubit é comumente representado pela notação de Dirac ou pela notação vetorial, ambas apresentadas na equação 2.1 (NIELSEN; CHUANG, 2010).

$$|\psi\rangle = \alpha_1 |0\rangle + \alpha_2 |1\rangle, \quad |\psi\rangle = \begin{bmatrix} \alpha_1 \\ \alpha_2 \end{bmatrix} \quad (2.1)$$

Na equação 2.1, os termos  $|0\rangle$  e  $|1\rangle$  representam os estados da base e a natureza probabilística do qubit é capturada pelas amplitudes complexas  $\alpha_1$  e  $\alpha_2$ . Até ser medido, o qubit no estado  $|\psi\rangle$  existe como a superposição dos estados da base com probabilidades  $|\alpha_1|^2$  para  $|0\rangle$  e  $|\alpha_2|^2$  para  $|1\rangle$ .

#### 2.1.1 Evolução de Estados

Um sistema quântico fechado evolui através de transformações unitárias (NIELSEN; CHUANG, 2010). A notação algébrica utilizada para descrever uma evolução do estado  $|\psi\rangle$  pelo operador unitário  $U$  é apresentada na equação 2.2.

$$|\psi'\rangle = U |\psi\rangle \quad (2.2)$$

É possível representar evoluções de forma gráfica utilizando a linguagem de circuitos. Nessa linguagem, as linhas representam os fios do circuito e conferem uma semântica de transcurso temporal ou mesmo as entidades físicas participativas do sistema. As portas quânticas são representadas por caixas e são responsáveis pela alteração do estado de forma equivalente aos operadores na notação algébrica (NIELSEN; CHUANG, 2010).



Figura 1 – Exemplo de circuito quântico

A figura 1 ilustra um circuito quântico inicializado em  $|0\rangle$  no qual é aplicada uma porta  $X$  seguida de um operador de medida.

A tabela 1 exibe alguns dos operadores mais comuns. Qualquer operação unitária em um único qubit pode ser escrita a partir do conjunto de operadores  $\{RY, RZ\}$ . Qualquer unitária arbitrária pode ser construída através de operadores atuando em um único qubit e o operador  $CNOT$ .

Operador	Matriz	Descrição	Circuito
<b>Porta X</b>	$\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$	Inverte $ 0\rangle \leftrightarrow  1\rangle$ .	
<b>Porta Y</b>	$\begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}$	Combina inversão e fase; gera rotações no eixo $Y$ .	
<b>Porta Z</b>	$\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$	Aplica uma fase de $\pi$ ao estado $ 1\rangle$ .	
<b>Hadamard</b>	$\frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$	Cria superposição entre $ 0\rangle$ e $ 1\rangle$ .	
<b>CNOT</b>	$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$	Aplica $X$ no alvo se o controle for $ 1\rangle$ .	
<b>Rotação RX</b>	$\begin{bmatrix} \cos(\theta/2) & -i \sin(\theta/2) \\ -i \sin(\theta/2) & \cos(\theta/2) \end{bmatrix}$	Rotação em torno do eixo $X$ .	
<b>Rotação RY</b>	$\begin{bmatrix} \cos(\theta/2) & -\sin(\theta/2) \\ \sin(\theta/2) & \cos(\theta/2) \end{bmatrix}$	Rotação em torno do eixo $Y$ .	
<b>Rotação RZ</b>	$\begin{bmatrix} e^{-i\theta/2} & 0 \\ 0 & e^{i\theta/2} \end{bmatrix}$	Rotação em torno do eixo $Z$ .	
<b>Rotação Geral</b>	$\exp(-i\frac{\theta}{2} \vec{n} \cdot \vec{\sigma})$	Rotação de $\theta$ em torno do eixo $\vec{n}$ .	

Tabela 1 – Operadores quânticos com representações matriciais e circuitais.

### 2.1.2 Medida

Medições quânticas consistem na aplicação de um conjunto de operadores  $\{M_m\}$  que obedecem à relação  $\sum_m M_m^\dagger M_m = I$  e realizam um dos possíveis resultados  $m$  que podem ocorrer durante a experimentação. A probabilidade do resultado  $m$  ocorrer para um sistema

inicialmente no estado  $|\psi\rangle$  é dada pela equação 2.3 (NIELSEN; CHUANG, 2010).

$$p(m) = \langle \psi | M_m^\dagger M_m | \psi \rangle \quad (2.3)$$

O estado do sistema após a aplicação do operador de medida  $M_m$  é dado por:

$$|\psi'\rangle = \frac{M_m |\psi\rangle}{\sqrt{\langle \psi | M_m^\dagger M_m | \psi \rangle}} \quad (2.4)$$

Medidas projetivas são um caso especial em que, além de satisfazer a relação  $\sum_m M_m^\dagger M_m = I$ , os operadores do conjunto  $\{M_m\}$  são projetores ortogonais. Essas medidas são descritas por observáveis ( $\mathcal{O}$ ) com decomposição espectral dada pela equação 2.5.

$$\mathcal{O} = \sum_m m P_m \quad (2.5)$$

Em 2.5, os valores de  $m$  correspondem aos possíveis resultados de medida. Cada valor de  $m$  corresponde ao autovalor associado ao projetor  $P_m$  do observável  $\mathcal{O}$ . Nessa formulação, a probabilidade do resultado  $m$  para um sistema inicialmente no estado  $|\psi\rangle$  é descrita pela equação 2.6 (NIELSEN; CHUANG, 2010).

$$p(m) = \langle \psi | P_m | \psi \rangle \quad (2.6)$$

O valor esperado do observável  $\mathcal{O}$  é dado pela equação 2.7 (NIELSEN; CHUANG, 2010).

$$\mathbb{E}(\mathcal{O}) = \langle \psi | \mathcal{O} | \psi \rangle \quad (2.7)$$

## 2.2 ALGORITMOS QUÂNTICOS VARIACIONAIS

Algoritmos Quânticos Variacionais são estruturas de processamento híbrido clássico-quântica em que um otimizador clássico é utilizado para treinar um circuito quântico parametrizado (CEREZO et al., 2021).

Um circuito quântico parametrizado pode ser definido como uma operação unitária ajustável em um sistema de  $n$  qubits. Esse tipo de circuito é o elo do processamento híbrido uma vez que estabelece a conexão entre recursos computacionais de natureza quântica e clássica (SIM; JOHNSON; ASPURU-GUZI, 2019).

De forma geral, a essência de um algoritmo quântico variacional apresenta 4 passos (QI et al., 2024):

1. Codificar um problema como uma função de custo  $\mathcal{C}(\theta)$  otimizável;

2. Construir um circuito quântico com parâmetros  $\theta$ ;
3. Medir o estado de saída e calcular a função de custo  $\mathcal{C}(\theta)$ ;
4. Atualizar  $\theta$  utilizando um otimizador clássico.

As subseções a seguir detalharão as ferramentas essenciais para o funcionamento de uma arquitetura de algoritmo quântico variacional.

### 2.2.1 Função de Custo

Uma função de custo é um mapeamento de um conjunto de parâmetros  $\theta$  para um número real. No contexto de algoritmos variacionais, ela é uma função de um conjunto de estados de entrada  $\{\rho_k\}$ , um conjunto de observáveis  $\{\mathcal{O}_k\}$  e um circuito parametrizado  $U(\theta)$ . Matematicamente (CEREZO et al., 2021):

$$\mathcal{C}(\theta) = f(\{\rho_k\}, \{\mathcal{O}_k\}, U(\theta)) \quad (2.8)$$

É desejável que o valor mínimo da função de custo coincida com a solução do problema a ser tratado por meio da abordagem variacional.

### 2.2.2 Ansatz

Circuitos quânticos parametrizados também são conhecidos como ansatz na literatura de algoritmos variacionais. Eles carregam a estrutura a ser atualizada a cada passo de otimização e podem ser matematicamente descritos como um produto de  $L$  operações unitárias sequencias (CEREZO et al., 2021):

$$U(\theta) = U_L(\theta_L) \cdots U_2(\theta_2)U_1(\theta_1) \quad (2.9)$$

O sucesso de um algoritmo variacional está fortemente associado à escolha do ansatz. A quantidade de parâmetros, a profundidade do circuito e sua capacidade de representar o espaço de soluções são aspectos importantes para embasar o uso de um ansatz em um algoritmo variacional (SIM; JOHNSON; ASPURU-GUZI, 2019). Alguns tipos de ansatz levam em consideração informações sobre o problema a ser resolvido, outros procuram atingir o requisito de baixa profundidade levando em consideração o hardware disponível (CEREZO et al., 2021). A figura 2 ilustra um exemplo de ansatz para um sistema de 3 qubits com portas de rotação  $R_Y$ , estruturas de emaranhamento com CNOT e parâmetros  $\{\theta_0, \theta_1, \theta_2, \theta_3\}$ .

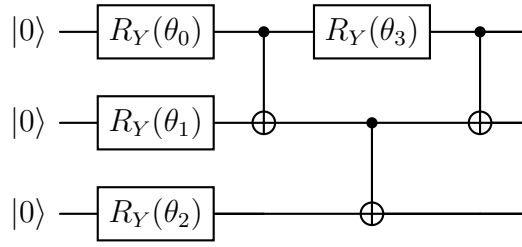


Figura 2 – Exemplo de ansatz

### 2.2.3 Gradientes

A saída do ansatz apresentado de forma genérica na subsecção 2.2.2 é obtida a partir de um conjunto de medições finita para extrair o valor esperado de um observável  $\mathcal{O}$ . As principais técnicas para obter a derivada parcial de  $\langle \mathcal{O} \rangle_{\theta}$  em relação à entrada  $\theta_i$  são (HARROW; NAPP, 2021):

- **Diferenças Finitas:** a derivada parcial é obtida utilizando apenas as medições de  $\langle \mathcal{O} \rangle$  com  $\theta_i$  deslocado de um pequeno valor  $\epsilon$  por meio da expressão 2.10.

$$\frac{\partial \langle \mathcal{O} \rangle_{\theta}}{\partial \theta_i} \approx \frac{1}{2\epsilon} [\langle \mathcal{O} \rangle_{\theta + \epsilon \hat{e}_i} - \langle \mathcal{O} \rangle_{\theta - \epsilon \hat{e}_i}] \quad (2.10)$$

- **Medições Analíticas:** a derivada parcial  $\frac{\partial \langle \mathcal{O} \rangle_{\theta}}{\partial \theta_i}$  é extraída do dispositivo quântico. Nessa formulação,  $\frac{\partial \langle \mathcal{O} \rangle_{\theta}}{\partial \theta_i}$  é expressa em parcelas que podem ser obtidas a partir de medições no processador quântico e posteriormente combinadas em um coprocessador clássico (SCHULD et al., 2019).

Algoritmos variacionais baseados em medições analíticas apresentam melhores taxas de convergência (HARROW; NAPP, 2021). No contexto de aprendizagem variacional, o uso de medições analíticas para obter gradientes melhora o desempenho do otimizador (LOCKWOOD, 2022).

### 2.2.4 Otimização

Algoritmos variacionais otimizam classicamente uma função de custo definida em termos do parâmetro  $\theta$  da equação 2.9. É possível agrupar as técnicas de otimização em duas categorias: as que fazem uso do gradiente descendente e as que não fazem (CEREZO et al., 2021).

Abordagens baseadas no gradiente descendente atualizam o parâmetro  $\theta$  iterativamente na direção contrária do gradiente da função de custo  $\mathcal{C}(\theta)$  ponderado por um fator  $\alpha$ . O passo

iterativo de atualização é dado pela equação 2.11.

$$\boldsymbol{\theta}^{t+1} \leftarrow \boldsymbol{\theta}^t - \alpha \nabla \mathcal{C}(\boldsymbol{\theta}) \quad (2.11)$$

O gradiente  $\nabla \mathcal{C}(\boldsymbol{\theta})$  em 2.11 pode ser obtido a partir dos métodos apresentados na subseção 2.2.3.

## 2.3 APRENDIZAGEM DE MÁQUINAS

Uma definição de aprendizagem de máquinas consiste em considerar uma classe de tarefas  $T$ , uma medida de performance  $P$  e uma experiência  $E$ . Com base nesses elementos, a aprendizagem consiste em fazer com que a performance de uma máquina na classe de tarefas  $T$ , medida por  $P$ , melhore com a experiência  $E$  (MITCHELL, 1997).

A forma como a experiência  $E$  é estruturada conduz a diferentes tipos de aprendizagem. Um dos tipos é o supervisionado em que a dinâmica de aprendizagem é assistida por um professor (GOODFELLOW et al., 2016).

Algoritmos de aprendizagem supervisionada recebem como entrada conjuntos de dados rotulados e tem por objetivo encontrar um relacionamento entre características e rótulos. As características são informações acerca do domínio do problema. Os rótulos são valores inseridos por um professor ou especialista do domínio.

### 2.3.1 Classificação Binária

Um problema de classificação é um exemplo de aprendizagem supervisionada. Esse tipo de problema assume como premissa a existência de uma relação entre entrada e saída nos dados disponíveis. A busca desse relacionamento pode ser expressa matematicamente como selecionar uma função  $h$  em um conjunto de hipóteses  $\mathcal{H}$  que obtenha melhor resultado em associar o elemento  $\mathbf{x}_i$  à saída  $y_i$  do conjunto  $\mathcal{D} = \{(\mathbf{x}_i, y_i), i = 1, 2, \dots, N\}$ .

Para aferir a qualidade de um resultado é necessária uma métrica  $\mathcal{L}$  que indique de forma quantitativa a taxa de aderência de um modelo  $h$  aos dados  $\mathcal{D}$ . A métrica  $\mathcal{L}$  recebe como parâmetros  $h(\mathbf{x}_i)$  e  $y_i$  e retorna um valor representativo do quanto o resultado de  $h(\mathbf{x}_i)$  está próximo da saída  $y_i$ .

Uma das formas de selecionar a hipótese  $h \in \mathcal{H}$  é assumir uma forma paramétrica  $h_{\boldsymbol{\theta}}$  onde  $\boldsymbol{\theta} \in \mathbb{R}^d$ . O problema de aprendizagem se resume a encontrar uma configuração paramétrica

que minimize o erro de classificação. Se considerarmos  $\mathcal{C}(\boldsymbol{\theta}) = \frac{1}{N} \sum_{i=1}^N \mathcal{L}(h_{\boldsymbol{\theta}}(\mathbf{x}_i), y_i)$  como o erro associado à entrada paramétrica  $\boldsymbol{\theta}$ , é possível definir matematicamente um problema de aprendizagem como:

$$\min_{\boldsymbol{\theta}} \mathcal{C}(\boldsymbol{\theta}) = \min_{\boldsymbol{\theta}} \frac{1}{N} \sum_{i=1}^N \mathcal{L}(h_{\boldsymbol{\theta}}(\mathbf{x}_i), y_i) \quad (2.12)$$

A depender do domínio da saída, um problema de classificação pode ser binário ou multi-classe. No caso de classificação binária, a saída assume apenas dois valores distintos. No caso multiclasse, a saída pode assumir mais de dois valores.

### 2.3.2 Classificadores Quânticos

Algoritmos Quânticos Variacionais podem ser utilizados em tarefas de aprendizagem de máquinas. Nessa abordagem, a camada variacional descrita na subseção 2.2.2 é interpretada como modelo e o processo de aprendizagem acontece iterativamente através do ajuste paramétrico para minimizar o custo (BENEDETTI et al., 2019). Modelos de aprendizagem construídos com base em algoritmos variacionais são aproximadores universais de funções (SCHULD; SWEKE; MEYER, 2021).

Um classificador quântico é um caso particular de algoritmo variacional. Ele funciona de forma similar à abordagem clássica, substituindo a família paramétrica de hipóteses  $h_{\boldsymbol{\theta}} \in \mathcal{H}$  discutida na subseção anterior por um circuito parametrizado. Um classificador construído nesse formato possui 3 elementos básicos: uma estrutura codificadora, uma camada de processamento para evoluir estados e uma regra de decisão (LAROSE; COYLE, 2020).

#### 2.3.2.1 Codificação

A depender do tipo do dado, um modelo de aprendizagem quântica pode exigir um passo adicional em seu fluxo de execução. Quando o dado é de origem quântica, a exemplo dos dados provenientes de sensores quânticos, não há necessidade de uma etapa de codificação. Quando o dado é clássico, torna-se necessário representá-lo em um estado quântico para processá-lo.

Algumas formas de representação mais comuns são: pela base, pela amplitude e pelo ângulo. A seguir serão detalhadas cada uma delas.



### 2.3.2.1.1 Base

A codificação por base associa o estado de cada qubit à representação binária de uma dada entrada. De forma geral, seja  $z$  um valor escalar com representação binária equivalente à mostrada na equação 2.13.

$$b = b_0 b_1 \dots b_{n-1} \quad (2.13)$$

O estado  $|b_0 b_1 \dots b_{n-1}\rangle$  é definido como a representação do valor escalar  $z$  pelo método de codificação pela base. Uma forma de preparar esse estado é considerar um sistema inicializado em  $|0\rangle$  com uma quantidade de qubits equivalente ao comprimento da string binária de  $z$  e aplicar portas  $X$  aos qubits onde a representação binária assume valor 1. A equação 2.14 apresenta o operador capaz de codificar um escalar  $z$  com representação binária  $b$  utilizando o método descrito.

$$\mathcal{E}(b) = \bigotimes_{i=0}^{n-1} X^{b_i} \quad (2.14)$$

Para um vetor  $\mathbf{x} = (x_0, x_1, \dots, x_{N-1})$  com  $N$  entradas escalares, a codificação pela base requer uma quantidade de qubits proporcional ao comprimento da string binária da entrada ( $n$ ). A complexidade de preparação do estado é  $O(Nn)$ .

### 2.3.2.1.2 Amplitude

A codificação por amplitude associa as amplitudes de um conjunto de qubits com as coordenadas de um vetor complexo normalizado. Matematicamente, se  $\mathbf{x} = (x_0, x_1, \dots, x_{N-1})$  é um vetor de  $\mathbb{C}^N$  com  $\sum_{i=0}^{N-1} |x_i|^2 = 1$ , o estado  $|\psi_{\mathbf{x}}\rangle$  codificado por amplitude para  $\mathbf{x}$  é dado pela equação 2.15.

$$|\psi_{\mathbf{x}}\rangle = \sum_{i=0}^{N-1} x_i |i\rangle \quad (2.15)$$

Existem diferentes propostas de implementação desse esquema de codificação com diferentes complexidades. Mottonen et al. (2004) propõem um método com complexidade exponencial no número de qubits. Araujo et al. (2021) apresentam um algoritmo com complexidade polilogarítmica no tamanho da entrada.

A codificação por amplitude faz uso de uma quantidade de qubits equivalente à  $\log N$ , onde  $N$  é a quantidade de amplitudes a serem carregadas. Sob determinadas condições, esse tipo de codificação apresenta complexidade  $O(N)/O(\log N)$  (SCHULD; PETRUCCIONE, 2021).

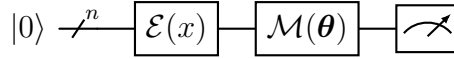


Figura 3 – Modelo de aprendizagem baseado em circuito quântico parametrizado

### 2.3.2.1.3 Ângulo

A codificação por ângulo associa cada entrada de um vetor  $\mathbf{x} = (x_0, x_1, \dots, x_{N-1})$  ao ângulo de uma porta de rotação aplicada em um qubit. A equação 2.16 ilustra um operador  $U$  que prepara a codificação descrita para o vetor  $\mathbf{x}$  em um sistema de  $N$  qubits aplicando portas de rotação em  $Y$ .

$$\mathcal{E}(\mathbf{x}) = \bigotimes_{i=0}^{N-1} R_Y(x_i) \quad (2.16)$$

A codificação por ângulo requer uma quantidade de qubits equivalente ao tamanho da entrada ( $N$ ). A complexidade de preparação do estado é  $O(N)$ .

### 2.3.2.2 Processamento

Modelos baseados em circuitos variacionais fazem uso de um circuito composto de duas partes, sendo uma destinada à codificação segundo alguma técnica como as apresentadas na subseção 2.3.2.1 e outra voltada ao modelo propriamente dito. A figura 3 ilustra um circuito variacional representativo dessa abordagem, onde  $\mathcal{E}(\mathbf{x})$  corresponde à uma estrutura de codificação para a amostra  $\mathbf{x}$  do conjunto de dados e  $\mathcal{M}(\boldsymbol{\theta})$  o modelo com parâmetros  $\boldsymbol{\theta}$  a serem ajustados.

O operador  $U(\mathbf{x}; \boldsymbol{\theta})$  que sintetiza a estrutura é mostrado na equação 2.17.

$$U(\mathbf{x}, \boldsymbol{\theta}) = \mathcal{M}(\boldsymbol{\theta})\mathcal{E}(\mathbf{x}) \quad (2.17)$$

Circuitos quânticos variacionais podem ser interpretados como modelos de aprendizagem determinísticos ou probabilísticos (SCHULD; PETRUCCIONE, 2021). O que diferencia uma abordagem da outra é o tratamento que se dá à saída do circuito. Essa distinção será abordada nas subseções que seguem.

#### 2.3.2.2.1 Modelos Quânticos Determinísticos

Um modelo determinístico se aproveita da estrutura da equação 2.17 e utiliza o valor esperado de um operador de medição  $\mathcal{O}$  como a saída do modelo (SCHULD; PETRUCCIONE,

2021). Matematicamente, se  $|\psi(\mathbf{x}, \boldsymbol{\theta})\rangle = U(\mathbf{x}, \boldsymbol{\theta}) |0\rangle$  é o estado resultante do circuito após a etapa de codificação e processamento do modelo, a saída  $f_{\boldsymbol{\theta}}(\mathbf{x})$  para o observável  $\mathcal{O}$  é dada por:

$$f_{\boldsymbol{\theta}}(\mathbf{x}) = \langle \psi(\mathbf{x}, \boldsymbol{\theta}) | \mathcal{O} | \psi(\mathbf{x}, \boldsymbol{\theta}) \rangle \quad (2.18)$$

### 2.3.2.2.2 Modelos Quânticos Probabilísticos

Diferentemente da abordagem determinística que utiliza o valor esperado de um operador de medição  $\mathcal{O}$  como a saída do modelo, modelos probabilísticos associam cada medição a um valor do domínio de saída  $y \in \mathcal{Y}$  (SCHULD; PETRUCCIONE, 2021). Nessa perspectiva,  $\mathcal{O}$  pode ser expresso em termos das possíveis saídas como  $\sum_{y \in \mathcal{Y}} y |y\rangle \langle y|$  e o modelo quântico probabilístico é definido pela equação 2.19.

$$p_{\boldsymbol{\theta}}(y|\mathbf{x}) = |\langle y | \psi(\mathbf{x}, \boldsymbol{\theta}) \rangle|^2 \quad (2.19)$$

A equação 2.19 mede a incerteza da saída  $y$  para uma dada entrada  $\mathbf{x}$ . Um classificador pode utilizar a quantificação dessa incerteza para estabelecer mecanismos de predição.

### 2.3.2.3 Regra de Decisão

Algoritmos variacionais exibem saídas desprovidas de uma relação semântica com a atividade fim de um classificador. A regra de decisão é o componente responsável por estabelecer um relacionamento semântico entre as saídas dos processamentos e o critério de classificação (LAROSE; COYLE, 2020).

Em problemas de classificação binária é usual selecionar apenas um qubit para fazer medições e extrair o valor esperado de um observável (SCHULD et al., 2020). Ao valor esperado obtido é adicionado um viés treinável e o resultado é mapeado em um conjunto com dois valores. A função responsável pelo mapeamento estabelece o critério de decisão.

## 2.4 CONSIDERAÇÕES FINAIS

O presente capítulo abordou um recorte multidisciplinar dos conhecimentos que subsidiam os desenvolvimentos posteriores deste trabalho. A seção 2.1 apresentou o modelo de computação utilizado, mostrando como a unidade básica de informação (qubit) é manipulada por

transformações unitárias e como a informação é extraída através de medições. A seção 2.2 detalhou a estrutura e funcionamento do modelo algorítmico variacional, destacando considerações sobre a eficiência relativas à obtenção dos gradientes e ao desempenho do otimizador. Por fim, a seção 2.3 abordou a empregabilidade dos algoritmos variacionais no contexto de aprendizagem supervisionada, apresentando a formulação de um classificador binário que tem como entrada dados clássicos codificados em estados quânticos e cuja saída pode ser interpretada por diferentes mecanismos.

### 3 BATCH QUÂNTICO

#### 3.1 BATCH CLÁSSICO

O problema de minimização da função de custo descrito em 2.12 pode ser resolvido de forma iterativa utilizando o conjunto de dados  $\mathcal{D} = \{(\mathbf{x}_i, y_i), i = 1, 2, \dots, N\}$  e o algoritmo do Gradiente Descendente. Esse algoritmo atualiza o parâmetro  $\boldsymbol{\theta}$  na direção contrária ao gradiente do custo ( $\nabla \mathcal{C}(\boldsymbol{\theta})$ ) ponderado pelo fator  $\eta$ :

$$\boldsymbol{\theta}^{t+1} \leftarrow \boldsymbol{\theta}^t - \eta \nabla \mathcal{C}(\boldsymbol{\theta}^t) \quad (3.1)$$

Em virtude do fator  $\nabla \mathcal{C}(\boldsymbol{\theta})$  em 3.1 ser calculado para todo o conjunto  $\mathcal{D}$ , a cardinalidade desse conjunto pode afetar de forma expressiva o desempenho do algoritmo (BOTTOU et al., 1991). Uma das formas de superar essa limitação é fazer atualizações paramétricas a cada nova apresentação de um exemplo  $(\mathbf{x}_i, y_i) \in \mathcal{D}$  escolhido de forma aleatória. Essa estratégia de atualização reduz em um fator de  $N$  o cômputo de  $\nabla \mathcal{C}(\boldsymbol{\theta})$  em 3.1 e é conhecida como uma versão estocástica do Gradiente Descendente.

Outra estratégia de atualização paramétrica é considerar subconjuntos de  $\mathcal{D}$  com cardinalidade  $W$  e calcular o gradiente para cada um desses subconjuntos. Uma escolha razoável da quantidade de subconjuntos torna menos custosa a determinação do gradiente. É possível, ainda, encontrar estratégias que relacionam  $W$  com outros hiperparâmetros. He, Liu e Tao (2019) trazem evidências teóricas e empíricas de uma relação de  $W$  com a taxa de aprendizagem de forma a ampliar a capacidade de generalização em modelos de aprendizagem profunda.

O conceito de batch em aprendizagem clássica está intrinsecamente relacionado às estratégias escolhidas para fazer atualizações paramétricas do tipo 3.1. Quando todo o conjunto  $\mathcal{D}$  é utilizado para calcular o gradiente, tem-se uma aprendizagem em batch. Quando apenas uma amostra em  $\mathcal{D}$  é utilizada, tem-se uma aprendizagem online. Um aspecto vantajoso desse último tipo de aprendizagem é a capacidade de atualizar o modelo a medida que novas amostras de treino estão disponíveis, conferindo instantaneidade à sua forma de operação (HAYKIN, 2008).

## 3.2 O EQUIVALENTE QUÂNTICO

Estratégias de aprendizagem em batch exploram formas diferentes de apresentar padrões a algoritmos como foi detalhado na seção 3.1. As subseções a seguir irão explorar novas estratégias de batch em padrões codificados em estados quânticos.

### 3.2.1 Batch Sequencial

Consiste em apresentar um ou mais padrões codificados em estados quânticos e apresentar um de cada vez. O único diferencial dessa abordagem é utilizar o mapeamento de dados clássicos em quânticos e explorar as possíveis correlações existentes em um novo espaço para obter resultados melhores. A figura 4 ilustra um batch sequencial de tamanho 4. Cada  $|\psi\rangle_{i \in \{1,2,3,4\}}$  corresponde a um padrão carregado e depois processado por uma camada variacional ( $\mathcal{M}(\theta)$ ).

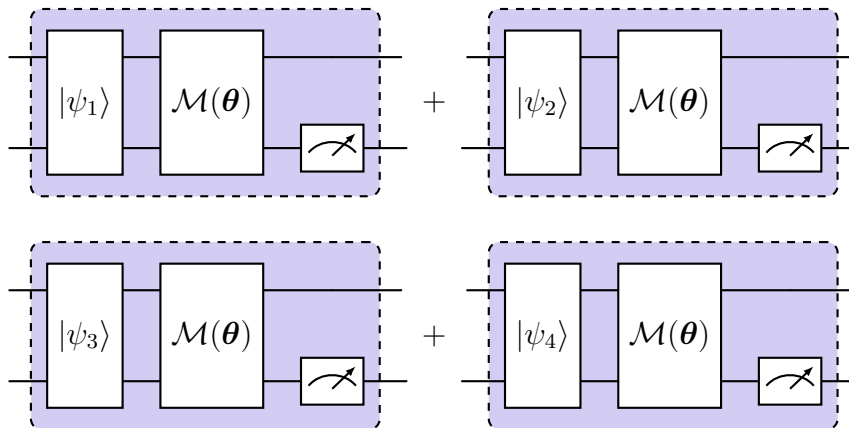


Figura 4 – Exemplo de batch sequencial de tamanho 4

Para um circuito variacional descrito pela equação 2.17, o custo  $\mathcal{C}(\theta)$  pode ser descrito em termos da função de perda  $\mathcal{L}$ , das entradas  $\mathbf{x}_i$  e do parâmetro variacional  $\theta$ , para um batch de tamanho  $n$ , como:

$$\mathcal{C}(\theta) = \frac{1}{n} \sum_{i=1}^n \mathcal{L}(U(\mathbf{x}_i; \theta), y_i) \quad (3.2)$$

### 3.2.2 Batch Superposição

Consiste em codificar 2 ou mais padrões em estados quânticos e colocá-los em superposição. A figura 5 ilustra um batch superposição com 4 padrões carregados em estados

$|\psi\rangle_{i \in \{1,2,3,4\}}$ , colocados em superposição e processados por uma camada variacional ( $\mathcal{M}(\theta)$ ).

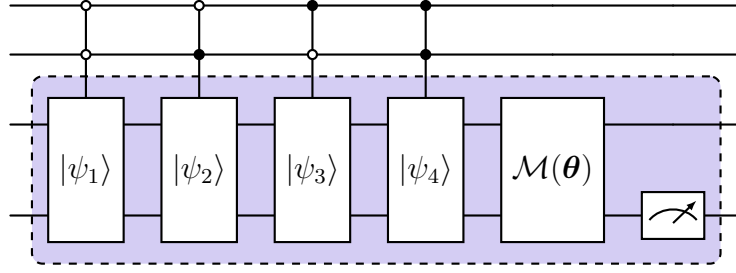


Figura 5 – Exemplo de batch sequencial de tamanho 4

Para o batch em superposição, a expressão da equação 3.2 se reduz a:

$$\mathcal{C}(\theta) = \mathcal{L}(U(\mathbf{x}_1, \dots, \mathbf{x}_n; \theta), y_i) \quad (3.3)$$

O circuito que implementa um batch em superposição é composto de 2 subsistemas. Um deles serve para criar uma camada de endereçamento dos estados a serem carregados no outro subsistema. A figura 6 ilustra um circuito com o caso geral de um batch em superposição em que  $e$  denota a quantidade de qubits no subsistema de endereçamento,  $d$  representa a quantidade de qubits no subsistema destinado ao carregamento dos dados,  $\mathcal{E}$  o operador responsável por carregar dados clássicos no dispositivo quântico e  $\mathcal{M}(\theta)$  é um circuito parametrizado em  $\theta$ .

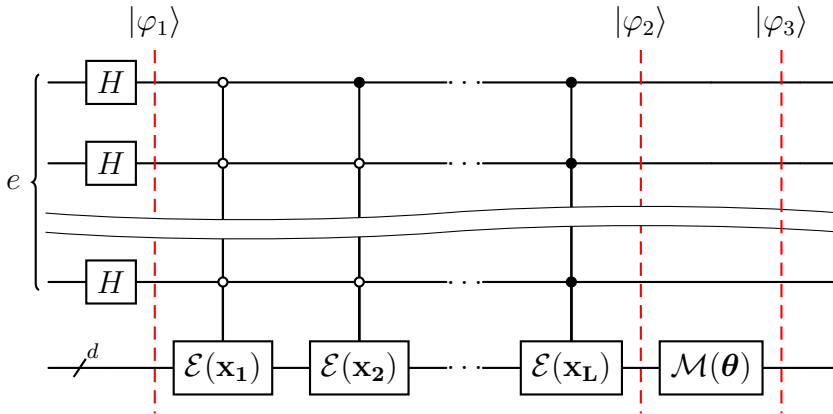


Figura 6 – Batch Superposição - caso geral

Assumindo um estado inicial  $|\varphi_0\rangle = |0\rangle^{\otimes(e+d)}$ , o primeiro passo para construir um batch em superposição é aplicar portas Hadamard ao subsistema de endereçamento. O estado do sistema, representado por  $|\varphi_1\rangle$  na figura 6, é dado pela expressão em 3.4.

$$|\varphi_1\rangle = (H^{\otimes e} \otimes I^{\otimes d}) |\varphi_0\rangle = \frac{1}{\sqrt{2^e}} \left( \sum_{i \in \{0,1\}^e} |i\rangle \right) |0\rangle^{\otimes d} \quad (3.4)$$

A aposição de portas Hadamard nos qubits de endereçamento cria uma camada com a superposição de todos estados possíveis para a quantidade  $e$ , resultando em um total de  $2^e$ . O segundo passo para construir um batch em superposição é carregar os dados clássicos  $\mathbf{x}_i$  no dispositivo quântico, inserindo controles distintos de forma a esgotar a quantidade total de endereços. Dessa forma, a quantidade total de dados clássicos ( $L$ ) em um batch em superposição é limitada à quantidade de endereços disponíveis ( $L = 2^e$ ). O estado do sistema após o carregamento dos dados, representado por  $|\varphi_2\rangle$ , é dado pela expressão 3.5.

$$|\varphi_2\rangle = C^{(L-1)}(\mathcal{E}(\mathbf{x}_L)) \cdots C^1(\mathcal{E}(\mathbf{x}_2))C^0(\mathcal{E}(\mathbf{x}_1))|\varphi_1\rangle \quad (3.5)$$

Cada operador  $C^i$  denota a operação controlada nos  $e$  qubits de endereçamento e  $i$  o valor decimal do controle representado por uma string binária de comprimento  $e$ .

Combinando as expressões obtidas nas equações 3.4 e 3.5,  $|\varphi_2\rangle$  pode ser reescrito como:

$$|\varphi_2\rangle = \frac{1}{\sqrt{2^e}} \left( \sum_{i \in \{0,1\}^e} |i\rangle \mathcal{E}(\mathbf{x}_{i_d}) |0\rangle^d \right) \quad (3.6)$$

Na expressão 3.6, o subscrito  $i_d$  se refere à representação decimal de uma string binária  $i \in \{0,1\}^e$ . Chamemos de  $|\psi_{i_d}\rangle = \mathcal{E}(\mathbf{x}_{i_d}) |0\rangle^d$  o estado resultante da aplicação do operador  $\mathcal{E}$  ao dado clássico  $\mathbf{x}_{i_d}$  nos qubits de dados inicializados em  $|0\rangle$ . O estado final após a aplicação do circuito parametrizado  $\mathcal{M}(\boldsymbol{\theta})$ , representado por  $|\varphi_3\rangle$  na figura 6, é dado por:

$$|\varphi_3\rangle = (I^{\otimes e} \otimes \mathcal{M}(\boldsymbol{\theta})) \frac{1}{\sqrt{2^e}} \left( \sum_{i \in \{0,1\}^e} |i\rangle |\psi_{i_d}\rangle \right) = \frac{1}{\sqrt{2^e}} \left( \sum_{i \in \{0,1\}^e} |i\rangle \mathcal{M}(\boldsymbol{\theta}) |\psi_{i_d}\rangle \right) \quad (3.7)$$

A expressão da equação 3.7 indica que, após aplicar o circuito  $\mathcal{M}(\boldsymbol{\theta})$  que atua apenas nos qubits de dados, o estado final composto pelos dois subsistemas resulta na superposição do modelo aplicado a cada ponto de dado  $\mathbf{x}_{i_d}$  inicializado em um estado quântico pelo operador  $\mathcal{E}$ . Nesse ponto, ao selecionar um qubit de dados para medição, o sistema colapsa de forma aleatória para um dos valores contidos na superposição. Dessa forma, o valor esperado do observável aferido é composto uniformemente por contribuições advindas de todos os pontos de dados carregados.

Após a aplicação de um observável  $\mathcal{O}$  aos qubits de dados, o valor esperado da medida pode ser o obtido por meio da equação 3.8.

$$\mathbb{E}(\mathcal{O}) = \frac{1}{2^e} \left( \sum_{i \in \{0,1\}^e} \langle \psi_{i_d} | \mathcal{M}^\dagger(\boldsymbol{\theta}) \mathcal{O} \mathcal{M}(\boldsymbol{\theta}) | \psi_{i_d} \rangle \right) \quad (3.8)$$

As principais desvantagens do batch em superposição comparadas ao sequencial são o uso de mais espaço e a maior complexidade na preparação dos multicontrolados para incluir os qubits de endereçamento.



### 3.2.3 Batch Duplo

Consiste em uma abordagem híbrida que mescla os conceitos apresentados nas subseções anteriores. Cada batch duplo apresenta sequencialmente padrões codificados em superposições distintas. A figura 7 ilustra um batch duplo com 2 superposições distintas e 2 padrões em cada uma delas.

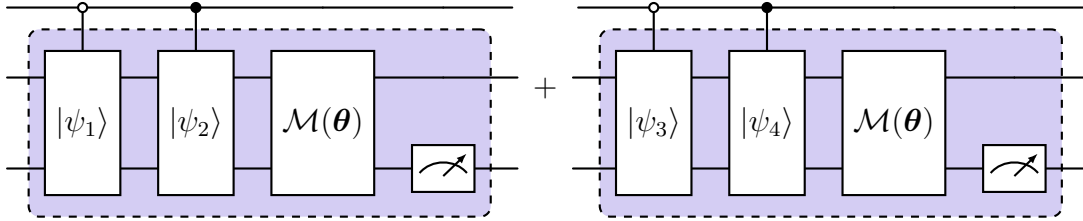


Figura 7 – Exemplo de batch duplo de tamanho 2 com superposições de tamanho 2

Para o batch duplo, a expressão do custo ( $\mathcal{C}$ ) faz uso de uma estrutura de soma no tamanho do batch sequencial como na equação 3.2 e de uma estrutura de carregamento de padrões em superposição como em 3.3. Matematicamente, para uma quantidade  $b$  de superposições distintas, cada uma com  $k$  padrões, o erro pode ser expresso como:

$$\mathcal{C}(\theta) = \sum_{j=1}^b \mathcal{L}(U(\mathbf{x}_1^j, \dots, \mathbf{x}_k^j; \theta), y_i) \quad (3.9)$$

onde  $\mathbf{x}_i^j$  denota o padrão  $i$  carregado na superposição  $j$ .

Expressões similares à de saída do batch em superposição podem ser obtidas para os circuitos componentes do batch duplo. Diferentemente das demais estratégias, a atualização de pesos no batch duplo soma classicamente os valores extraídos de dados em superposição. O aspecto atrativo em considerar esse tipo de abordagem é a economia que pode ser obtida na preparação dos estados sem abrir mão do recurso de superposição.

## 3.3 CONSIDERAÇÕES FINAIS

Em aprendizagem clássica, a única forma de manipular a apresentação de dados em batch é através da seleção de subconjuntos de padrões do conjunto de treino. Na otimização baseada em gradiente, esses subconjuntos selecionados são utilizados para computar o gradiente da função de custo e atualizar iterativamente os pesos.

A computação quântica permite pensar uma nova forma de apresentar padrões a partir do recurso da superposição. Esse recurso torna possível carregar mais de um padrão e processar

todo o conjunto carregado em um único fluxo de execução, permutando o custo computacional de tempo por espaço.

A seção 3.2 discutiu as diferentes formas em que o recurso da superposição pode ser explorado no plano semântico de aprendizagem em batch. A subseção 3.2.1 apresentou um modelo de batch em que cada padrão é carregado em um circuito e em seguida as saídas são processadas classicamente para avaliar a função de custo. A subseção 3.2.2 apresentou um modelo em que todos os padrões são colocados em superposição e processados em um único circuito. A subseção 3.2.3 apresentou um modelo em que padrões são carregados em superposições distintas. O que diferencia os modelos apresentados é a quantidade de parcelas no cômputo da função de custo para estimar iterativamente o gradiente.

## 4 METODOLOGIA

Este capítulo descreve o processo experimental utilizado para avaliar o desempenho de um modelo variacional treinado com as diferentes estratégias de batch apresentadas no capítulo 3. Foram empregados batches de classe única de forma a suprimir o carregamento dos rótulos e do circuito comparativo entre o rótulo e o valor de predição, simplificando a arquitetura do modelo em termos de espaço. O modelo variacional foi fixado em duas camadas para manter baixa profundidade dos circuitos e isolar o efeito das estratégias de batch no treinamento.

A análise comparativa privilegiou problemas de classificação binária em virtude da simplicidade e aplicabilidade em diversos domínios práticos como detecção de fraudes e diagnósticos médicos. Sendo o foco do presente estudo explorar novas formas de treinamento em batch no domínio quântico, a comparação com modelos clássicos no estado da arte foi suprimida.

### 4.1 PROCESSAMENTO QUÂNTICO

#### 4.1.1 Entrada

Cada amostra foi codificada nas amplitudes para garantir maior economia de espaço no carregamento dos dados clássicos. Os batches de superposição foram construídos utilizando qubits auxiliares com portas Hadamard e operações controladas tendo como alvo as amostras codificadas. A figura 8 ilustra um circuito codificador do batch em superposição para 4 amostras. Cada  $|\psi_i\rangle_{i \in \{1,2,3,4\}}$  representa uma amostra carregada no estado  $|\psi_i\rangle$ .

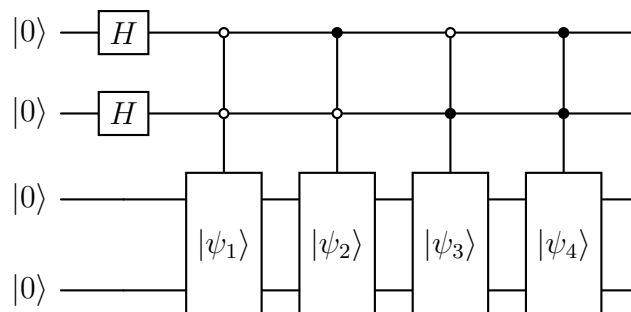


Figura 8 – Exemplo de entrada com 4 amostras carregadas em superposição

### 4.1.2 Modelo

O modelo utilizado em todos os experimentos foi construído de forma a manter baixa profundidade no circuito e com portas parametrizadas atuando em um ou dois qubits, conforme descrito em (SCHULD et al., 2020). As características de baixa profundidade e portas sob no máximo dois qubits permitem encurtar o tempo de experimentação sem comprometer o espaço de expressividade do modelo.

A arquitetura foi fixada com 2 camadas, cada uma com operações de rotação e uma estrutura de emaranhamento com portas CNOT, conforme exemplificado na figura para um sistema com 3 qubits.

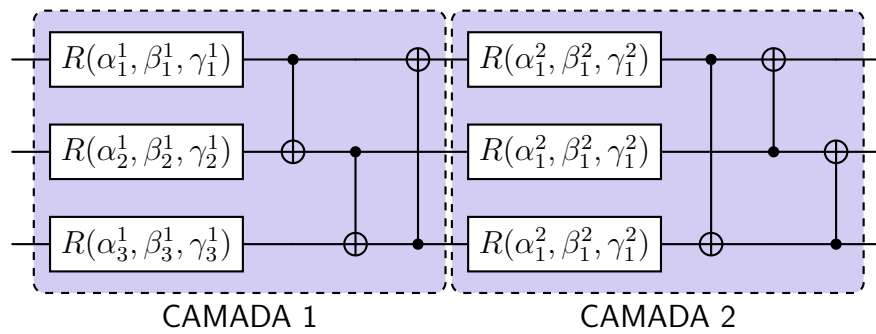


Figura 9 – Exemplo de modelo com 3 qubits e 2 camadas

Esse tipo de arquitetura usa uma quantidade de parâmetros treináveis de complexidade polilogarítmica no tamanho da entrada e apresenta resultados competitivos em benchmarks de dados clássicos (SCHULD et al., 2020), justificando sua adoção no presente trabalho.

### 4.1.3 Saída

O circuito correspondente aos esquemas de codificação apresentados em 4.1.1 acrescido da arquitetura de modelo especificada em 4.1.2 foi executado  $s$  vezes, sendo  $s$  um hiperparâmetro selecionado a depender da estratégia de batch. Para todas as execuções, foi fixado um qubit de dados para ser medido com o operador Pauli Z e o valor esperado extraído após as  $s$  execuções. O valor esperado de Pauli Z acrescido de um bias treinável foi utilizado para alimentar a regra de decisão do modelo.

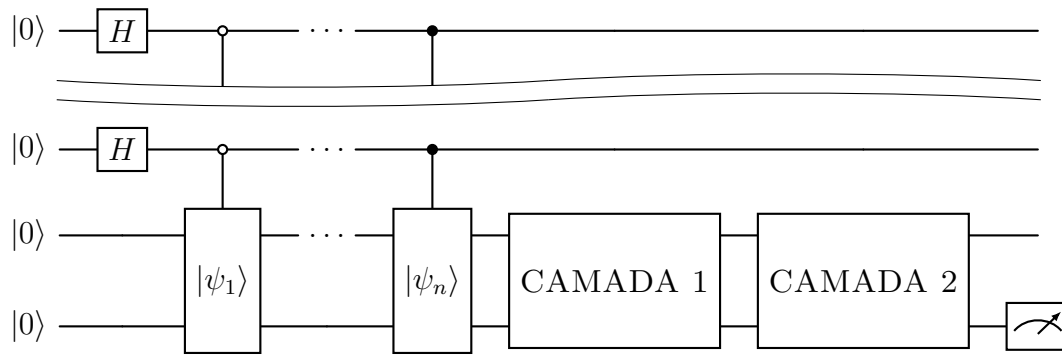


Figura 10 – Exemplo de circuito com entrada, modelo e medição

## 4.2 CONFIGURAÇÃO EXPERIMENTAL

### 4.2.1 Bases de Dados

A experimentação com as estratégias de batch discutidas no capítulo 3 foi feita com as bases IRIS e MNIST. Essas bases foram escolhidas por apresentarem diferentes níveis de complexidade e adequação ao aparato experimental.

A pequena quantidade de amostras e a baixa dimensionalidade da base IRIS a torna adequada para testes iniciais em circuitos de escala reduzida. A MNIST, sendo numerosa em amostras e dimensionalmente mais complexa, permite explorar limites de escalabilidade do aparato. Os tópicos a seguir irão detalhar cada uma delas, especificando o uso no corrente trabalho.

#### 4.2.1.1 IRIS

O conjunto de dados IRIS (FISHER, 1936) contém 150 amostras com registros métricos do comprimento e largura das sépalas e pétalas de 3 espécies de flores do tipo iris: versicolor, virginica e setosa. O conjunto está distribuído na proporção de 50 amostras para cada espécie e contém um rótulo adicional com a informação da espécie para cada registro métrico.

Para fins de classificação binária, foi utilizado um subconjunto da base IRIS com 100 amostras de flores do tipo setosa e versicolor, sendo 50 para cada tipo. A escolha desse subconjunto foi motivada pela existência de outros trabalhos na literatura que fazem uso do mesmo subconjunto, como em (PIRA; FERRIE, 2024) e em (DUAN; SUN; HSIEH, 2024). O conjunto de dados foi obtido na biblioteca scikit-learn (PEDREGOSA et al., 2011).

#### 4.2.1.2 MNIST

O conjunto de dados MNIST (LECUN et al., 1998) contém 70.000 amostras com imagens de dígitos de 0 à 9 escritos a mão. Cada imagem possui dimensão 28x28 pixels preenchidos com diversos níveis de cinza.

O subconjunto utilizado na experimentação fez uso de 400 amostras dos dígitos 3 e 6 em uma distribuição de 200 para cada. A escolha desse subconjunto foi motivada pela existência de outros trabalhos na literatura que fazem uso do mesmo subconjunto, como em (FARHI; NEVEN, 2018) e em (CHEN et al., 2020). O conjunto completo foi obtido na biblioteca keras (TEAM, 2024).

#### 4.2.2 Simulador Quântico

Toda a codificação dos experimentos foi feita utilizando PennyLane. O PennyLane é um framework de código aberto que contém uma série de recursos voltados para otimização de algoritmos quânticos e híbridos (BERGHOLM et al., 2018).

Foram utilizados os simuladores **default.qubit** e **lightning.gnu** para executar parte em estação de trabalho pessoal e no cluster Apuana, respectivamente. O **default.qubit** é um simulador baseado em Python ideal para prototipações de até 20 qubits. O **lightning.gpu** é um simulador escrito em C++ com otimização extremamente eficiente em dispositivos com 20 qubits ou mais.

#### 4.2.3 Pré-Processamento dos Dados

Os conjuntos de dados foram pré-processados com a normalização das características no intervalo  $[0, 1]$  e o mapeamento da saída no conjunto  $\{-1, 1\}$ . O propósito da normalização é carregar as características nas amplitudes obedecendo a condição de soma 1 das probabilidades dos estados. O mapeamento da saída foi feito para adequar o rótulo de cada padrão aos autovalores de medição do operador Pauli Z.

Além dos pré-processamentos descritos acima, o conjunto de dados MNIST foi redimensionado em 8x8 de forma a reduzir a quantidade de qubits utilizada durante a experimentação.

#### 4.2.4 Otimizador

A otimização dos parâmetros do modelo descrito na subseção 4.1.2 foi feita através do otimizador Adam com taxa de aprendizagem fixada em 0.01 (DIEDERIK, 2014). Esse otimizador é baseado no algoritmo do gradiente descendente apresentado no capítulo 3 e substitui o gradiente da equação 3.1 por uma expressão definida em termos de suas médias móveis para convergir mais rápido. Os gradientes dos circuitos foram obtidos utilizando a técnica de medições analíticas descrita na subseção 2.2.3. As implementações do otimizador e do gradiente foram obtidas na API do PennyLane (BERGHOLM et al., 2018).

#### 4.2.5 Métricas de Avaliação

A inexistência de métodos comparativos na literatura de aprendizagem quântica em batch tornou necessário o desenvolvimento de um critério valorativo apresentado ao longo dessa subseção.

O desempenho das estratégias descritas no capítulo 3 foi aferido a partir do conceito de densidade de shots no batch explicitado a seguir.

**Definição 1** *Sejam  $s$  e  $n$  as quantidades totais de shots e de padrões definidos para um batch, respectivamente. A densidade de shots ( $\mu$ ) no batch é dada pela relação:*

$$\mu = \frac{s}{n} \quad (4.1)$$

A motivação em construir essa definição é tornar os parâmetros  $s$  e  $n$  independentes da forma como cada estratégia os implementa. Para  $\mu = 8$  shots/padrão e  $n = 4$  padrões, por exemplo, a quantidade total de shots no batch é igual à 32. Cada uma das estratégias implementa um batch com  $\mu = 8$  de formas diferentes:

- Batch Sequencial: 4 circuitos, cada um com 1 padrão e 8 shots;
- Batch Superposição: 1 circuito com 4 padrões e 32 shots;
- Batch Duplo: 2 circuitos, cada um com 2 padrões e 16 shots.

A experimentação consistiu em fixar a densidade de shots em cada batch e comparar, para um mesmo valor de  $n$ , as estratégias em termos da quantidade de iterações até a convergência e das acurácias de treino e teste.

A escolha de fixar esse parâmetro foi motivada de forma a garantir que, independente da estratégia escolhida, cada batch contenha a mesma quantidade de padrões e que cada padrão seja apresentado a mesma quantidade de vezes a cada atualização de pesos do algoritmo de otimização.

Cada estratégia de batch emprega um ou mais circuitos para apresentar os padrões. Com a quantidade de padrões e a densidade de shots fixadas para o batch, torna-se necessário definir, para cada circuito, a quantidade de padrões a serem carregados e a respectiva quantidade de shots. Seja  $b$  a quantidade de circuitos utilizados por uma das estratégias apresentadas da seção 3.2. De forma genérica, a quantidade de padrões carregados ( $n_c$ ) e a quantidade de shots ( $s_c$ ) em cada circuito do batch são dados pelas expressões das equações 4.2 e 4.3.

$$n_c = \frac{n}{b} \quad (4.2)$$

$$s_c = \frac{s}{b} \quad (4.3)$$

As expressões quantitativas relativas aos padrões e shots por circuito apresentadas nas equações acima podem ser particularizadas para cada uma das estratégias definidas na seção 3.2. Assumindo  $n$  e  $\mu$  fixos, temos:

- **Batch Sequencial:** cada circuito carrega apenas um padrão, resultando em uma quantidade de circuitos equivalente a quantidade de padrões ( $b = n$ ). O número de padrões carregados no circuito ( $n_c$ ) é dado por  $n_c = \frac{n}{b} = \frac{n}{n} = 1$  e a quantidade de shots no circuito ( $s_c$ ) é dada por  $s_c = \frac{s}{b} = \frac{s}{n} = \mu$ ;
- **Batch Superposto:** um único circuito carrega todos os padrões em superposição, logo  $b = 1$ . Substituindo o valores de  $b$  nas equações 4.2 e 4.3, temos  $n_c = n$  e  $s_c = s$ ;
- **Batch Duplo:** esse é o caso mais geral e as expressões quantitativas referentes aos circuitos são definidas pelas equações 4.2 e 4.3.

Da análise dos valores de  $n_c$  e  $s_c$  apresentada acima, é possível constatar que as estratégias de batch sequencial e superposto são casos particulares da estratégia de batch duplo para os casos onde  $b$  assume os valores extremos de 1 e  $n$ . A tabela abaixo sintetiza as expressões quantitativas definidas para as estratégias detalhadas acima.



Estratégia	Sequencial	Superposto	Duplo
$n_c$	1	$n$	$1 < n_c < n$
$s_c$	$\mu$	$s$	$\mu < s_c < s$
$b$	$n$	1	$1 < b < n$

Tabela 2 – Síntese das expressões quantitativas de padrões e shots por circuito no batch

## 5 RESULTADOS

Os resultados comparativos foram gerados para a quantidade de padrões  $n$  assumindo valores no conjunto  $\{2, 4, 8\}$  e para a densidade de shots  $\mu$  em  $\{8, 16, 32\}$ , ambas as quantidades definidas por batch. O propósito de considerar quantidades de padrões múltiplas de 2 é carregar o maior valor possível de amostras pelo mecanismo de codificação discutido na subseção 4.1.1. As densidades de shots foram escolhidas de forma a garantir que cada padrão seja apresentado pelo menos uma vez para as diferentes quantidades de padrões pré-definidas. Para cada configuração  $(n, \mu)$  foram executados 10 experimentos e extraídos a média e o desvio.

As tabelas 3 e 4 exibem as configurações de hiperparâmetros para os valores de  $n$  e  $\mu$  fixados em cada estratégia.

(a) Hiperparâmetros Sequencial						(b) Hiperparâmetros Superposição					
Batch Sequencial				Circuito		Batch Superposição				Circuito	
n	$\mu$	s	b	$n_c$	$s_c$	n	$\mu$	s	b	$n_c$	$s_c$
2	8	16	2	1	8	2	8	16	1	2	16
	16	32	2	1	16		16	32	1	2	32
	32	64	2	1	32		32	64	1	2	64
4	8	32	4	1	8	4	8	32	1	4	32
	16	64	4	1	16		16	64	1	4	64
	32	128	4	1	32		32	128	1	4	128
8	8	64	8	1	8	8	8	64	1	8	64
	16	128	8	1	16		16	128	1	8	128
	32	256	8	1	32		32	256	1	8	256

Tabela 3 – Hiperparâmetros para as estratégias sequencial e superposição

As seções a seguir detalham os resultados obtidos para os conjuntos de dados IRIS e MNIST apresentados na subseção 4.2.1. Cada seção será segmentada pela quantidade de padrões ( $n$ ) e reunirá informações gráficas e numéricas para as estratégias de batch com diferentes configurações de hiperparâmetros.

Batch Duplo				Circuito	
n	$\mu$	s	b	n <sub>c</sub>	s <sub>c</sub>
4	8	32	2	2	16
	16	64	2	2	32
	32	128	2	2	64
8	8	64	2	4	32
			4	2	16
	16	128	2	4	64
			4	2	32
	32	256	2	4	128
			4	2	64

Tabela 4 – Hiperparâmetros para a estratégia de batch duplo

## 5.1 IRIS

### 5.1.1 $n = 2$

A análise dos gráficos da figura 11 permite constatar que a estratégia de batch em superposição apresenta melhor desempenho frente à estratégia sequencial para todos os valores de  $\mu$  analisados. É possível verificar ainda que a estratégia de batch em superposição apresenta convergência mais acelerada e curvas mais suaves.

A tabela 5 exibe os maiores valores das acurácias médias de treino e teste juntamente com a iteração em que foram obtidos. O melhor valor de acurácia média para o treino foi de 1.000 e ocorreu na iteração 144 da estratégia de batch em superposição com  $\mu = 32$  e apresentou desvio nulo. Para o valor médio da acurácia de teste, a estratégia de superposição com  $\mu = 16$  e  $\mu = 32$  juntamente com a estratégia sequencial para  $\mu = 32$  apresentaram valor médio de 1.000 com desvio nulo.

Hiperparâmetros				Treino			Teste		
n	$\mu$	b	Estratégia	Iteração1	Média1	Desvio1	Iteração2	Média2	Desvio2
2	8	2	Sequencial	153	0.895	0.037	205	0.910	0.066
		1	Superposição	221	<b>0.952</b>	0.017	161	<b>0.975</b>	0.040
	16	2	Sequencial	184	0.949	0.028	194	0.955	0.027
		1	Superposição	216	<b>0.991</b>	0.011	142	<b>1.000</b>	0.000
	32	2	Sequencial	228	0.988	0.010	141	<b>1.000</b>	0.000
		1	Superposição	144	<b>1.000</b>	0.000	85	<b>1.000</b>	0.000

Tabela 5 – Melhores acurácias de treino e teste para  $n = 2$

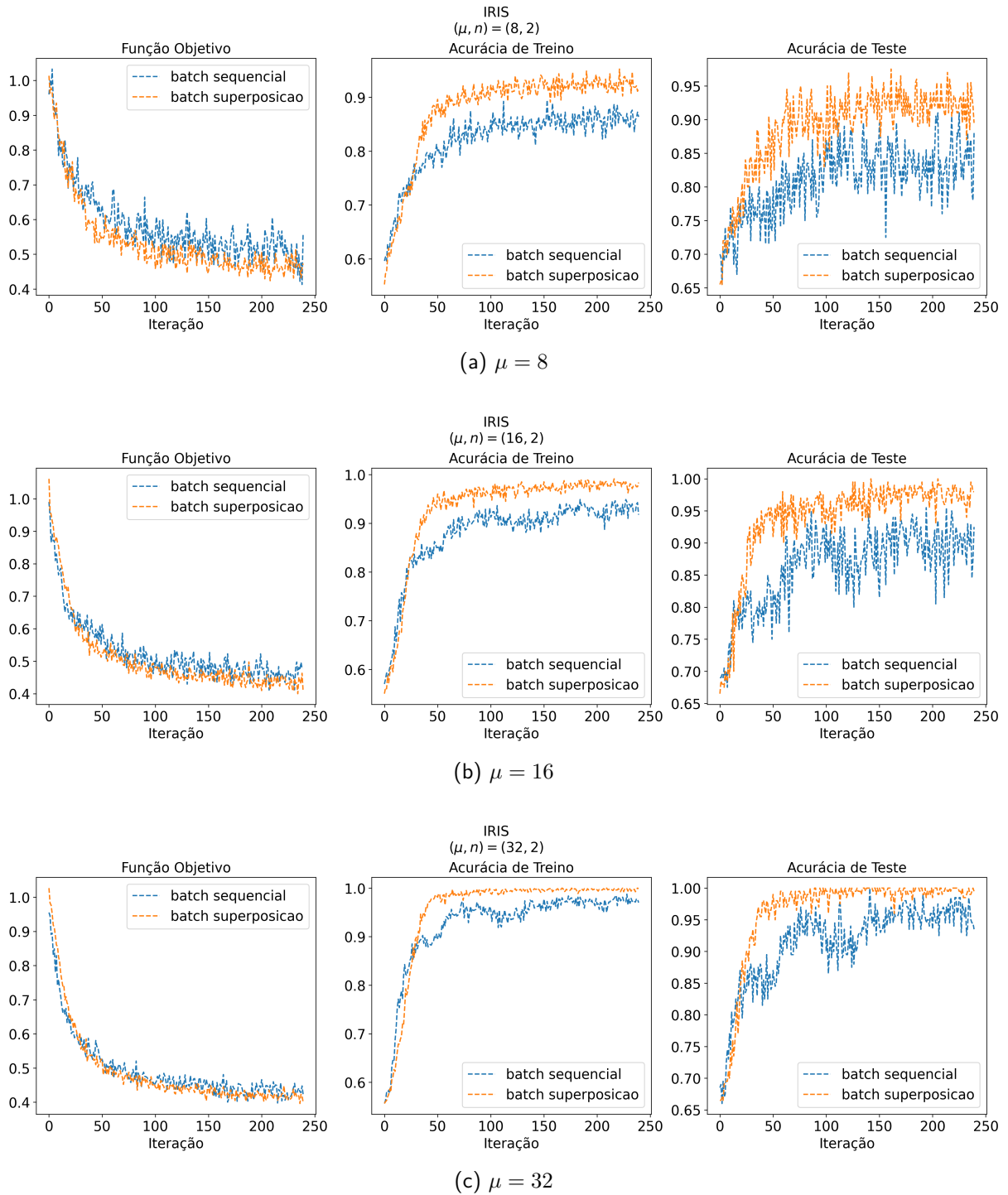


Figura 11 – Resultados IRIS para  $n = 2$

### 5.1.2 $n = 4$

A análise dos gráficos da figura 12 permite concluir que as estratégias de batch sequencial e de superposição apresentaram comportamentos similares ao caso  $n = 2$ . Para o batch duplo, o desempenho ficou intermediário entre as demais estratégias com uma relativa proximidade da estratégia de superposição a medida que a densidade de shots no batch aumenta. No gráfico

da figura 15c para  $\mu = 32$ , é possível identificar que as curvas de acurácia de treino e teste para as estratégias de batch em superposição e duplo ficam praticamente ajustadas.

A tabela 6 exibe os maiores valores das acurácias médias de treino e teste juntamente com a iteração em que foram obtidos. O melhor valor de acurácia média para o treino foi de 1.000 e ocorreu para as estratégias de batch em superposição e duplo para  $\mu \in \{16, 32\}$ . Para o valor médio da acurácia de teste, o melhor valor obtido foi 1.000 e ocorreu para as todas as estratégias para  $\mu \in \{16, 32\}$ . É possível constatar também que, para um mesmo valor de acurácia média atingido pelas 3 estratégias, a estratégia de batch em superposição alcança o valor máximo com uma quantidade menor de iterações.

Hiperparâmetros				Treino			Teste		
n	$\mu$	b	Estratégia	Iteração1	Média1	Desvio1	Iteração2	Média2	Desvio2
4	8	4	Sequencial	139	0.904	0.036	231	0.945	0.042
		1	Superposição	206	<b>0.991</b>	0.010	140	<b>1.000</b>	0.000
		2	Duplo	218	0.955	0.014	221	0.990	0.020
	16	4	Sequencial	224	0.954	0.021	163	0.985	0.032
		1	Superposição	106	<b>1.000</b>	0.000	80	<b>1.000</b>	0.000
		2	Duplo	223	0.992	0.011	234	<b>1.000</b>	0.000
	32	4	Sequencial	220	0.991	0.006	158	<b>1.000</b>	0.000
		1	Superposição	62	<b>1.000</b>	0.000	44	<b>1.000</b>	0.000
		2	Duplo	173	<b>1.000</b>	0.000	57	<b>1.000</b>	0.000

Tabela 6 – Melhores acurácias de treino e teste para  $n = 4$

### 5.1.3 $n = 8$

A análise dos gráficos da figura 13 permite extrair conclusões análogas às que foram apresentadas para os casos  $n = 2$  e  $n = 4$  nas subseções anteriores. O aspecto inovador da análise para o caso  $n = 8$  é o comportamento do hiperparâmetro  $b$  em diferentes estratégias e valores de  $\mu$ . Nas figuras 13a e 13b, valores menores de  $b$  possibilitam uma convergência mais acelerada com acurácias mais altas em um regime de baixa densidade de shots. O gráfico da figura 13c exibe um comportamento das curvas de acurácia praticamente sobrepostos para as estratégias de batch em superposição e duplo para os diferentes valores de  $b$ .

A tabela 7 exibe os maiores valores das acurácias médias de treino e teste juntamente com a iteração em que foram obtidos. O melhor valor de acurácia média para o treino foi de 1.000 e ocorreu para a estratégia de batch em superposição em todos os valores de  $\mu$  e para

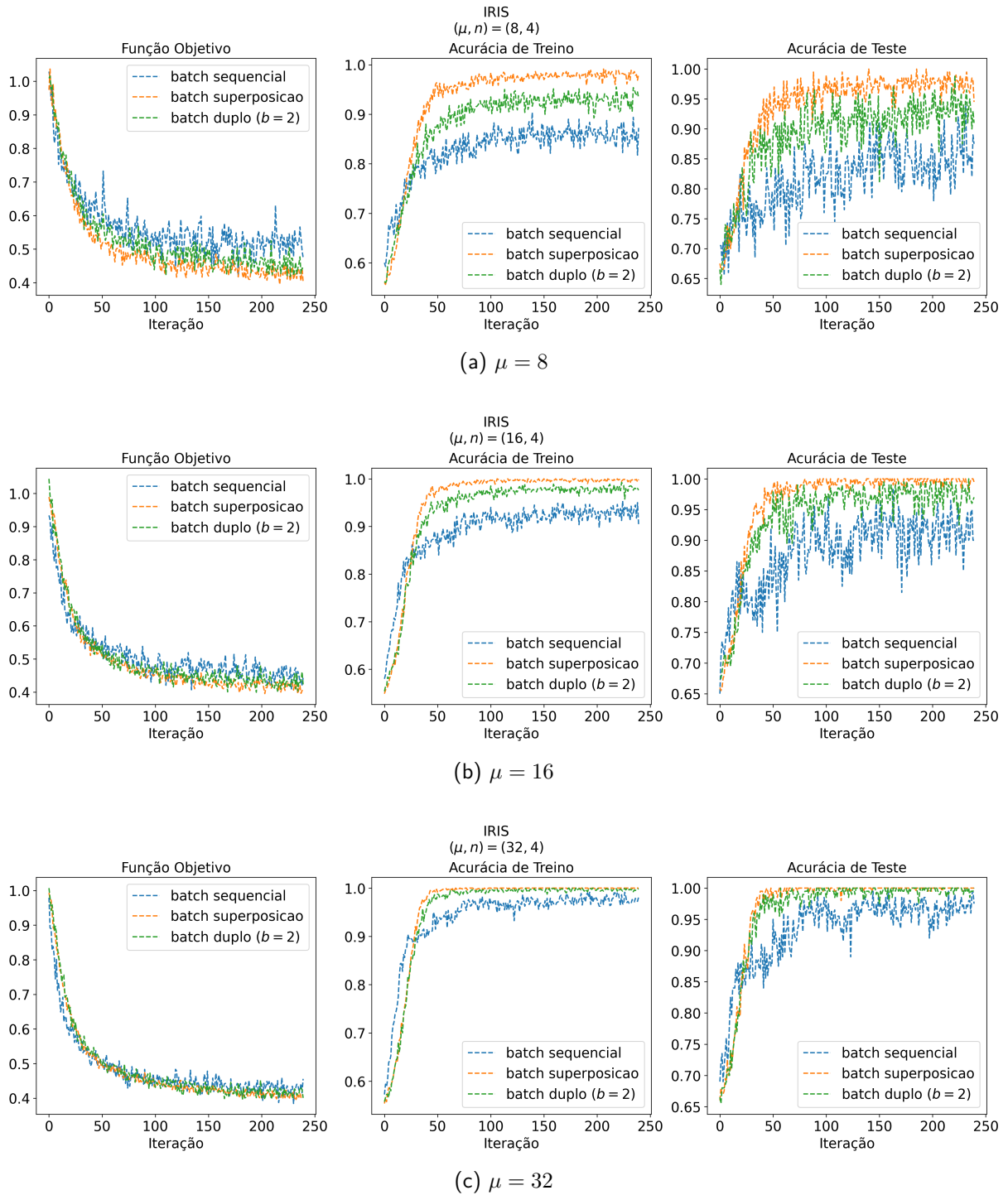


Figura 12 – Resultados IRIS para  $n = 4$

a estratégia de batch duplo com  $\mu \in \{16, 32\}$  em diferentes valores de  $b$ . Para o valor médio da acurácia de teste, o melhor valor obtido foi 1.000 e ocorreu para as estratégias de batch em superposição e duplo para  $\mu \in \{8, 16, 32\}$  e diferentes valores de  $b$ . É possível constatar também que, para um mesmo valor de acurácia média atingido pelas estratégias que fazem uso de padrões em superposição, valores menores de  $b$  conduzem uma quantidade menor de iterações para atingir o valor máximo. Diferentemente dos resultados exibidos nas tabelas 5

e 6, no caso  $n = 8$  a estratégia sequencial não atingiu o mesmo patamar das demais mesmo com o aumento da densidade de shots no batch. Isso sugere que a estratégia sequencial sofra algum tipo de perda em termos de poder de generalização com o aumento da quantidade de padrões no batch em configurações com baixo valor de  $\mu$ .

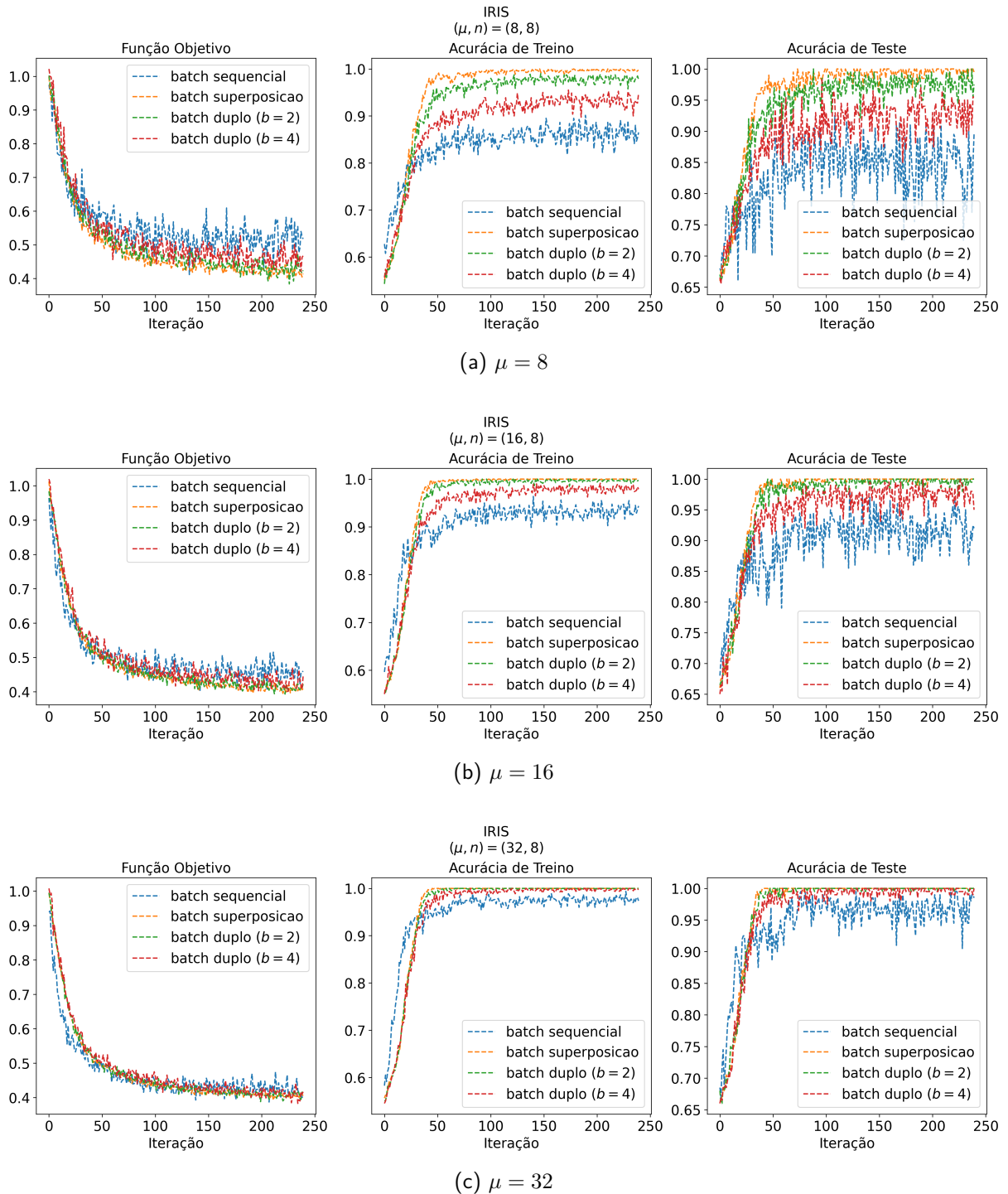


Figura 13 – Resultados IRIS para  $n = 8$

Hiperparâmetros				Treino			Teste		
n	$\mu$	b	Estratégia	Iteração1	Média1	Desvio1	Iteração2	Média2	Desvio2
8	8	8	Sequencial	189	0.901	0.028	108	0.925	0.056
		1	Superposição	146	<b>1.000</b>	0.000	73	<b>1.000</b>	0.000
		2	Duplo	147	0.990	0.009	88	<b>1.000</b>	0.000
		4	Duplo	173	0.955	0.024	96	0.985	0.023
	16	8	Sequencial	140	0.964	0.021	122	0.980	0.024
		1	Superposição	86	<b>1.000</b>	0.000	49	<b>1.000</b>	0.000
		2	Duplo	92	<b>1.000</b>	0.000	45	<b>1.000</b>	0.000
		4	Duplo	203	0.991	0.008	191	<b>1.000</b>	0.000
	32	8	Sequencial	227	0.994	0.006	76	0.995	0.015
		1	Superposição	45	<b>1.000</b>	0.000	40	<b>1.000</b>	0.000
		2	Duplo	58	<b>1.000</b>	0.000	47	<b>1.000</b>	0.000
		4	Duplo	81	<b>1.000</b>	0.000	52	<b>1.000</b>	0.000

Tabela 7 – Melhores acurácias de treino e teste para  $n = 8$ 

#### 5.1.4 Síntese dos Resultados

As análises apresentadas nas subseções 5.1.1, 5.1.2 e 5.1.3 permitem concluir que, fixados os hiperparâmetros  $(n, \mu)$  do batch, o ritmo de convergência do modelo e suas acurácias de treino e teste são afetados pelo hiperparâmetro  $b$  (quantidade de circuitos). Valores menores de  $b$  refletem uma convergência mais acelerada e acurácias maiores com menor desvio.

O aparato experimental selecionado obteve acurácias de treino e teste com valor 1.000 e desvio nulo para a estratégia de superposição em boa parte dos cenários avaliados. Para as demais estratégias, os valores de acurácia obtidos foram superiores ou próximos de 0.9 indicando que, a despeito da minimalidade dos recursos empregados na definição e operação do modelo, o aparato experimental conseguiu produzir resultados expressivos.

## 5.2 MNIST

### 5.2.1 $n = 2$

A análise dos gráficos da figura 14 permite extrair conclusões análogas às que foram apresentadas para o caso  $n = 2$  do conjunto de dados IRIS na seção anterior. O único diferencial para o conjunto MNIST diz respeito à suavidade das curvas.

A tabela 8 exhibe os maiores valores das acurácias médias de treino e teste juntamente



com a iteração em que foram obtidos. O melhor valor de acurácia média para o treino foi de 0.913 e ocorreu na iteração 887 da estratégia de batch em superposição com  $\mu = 32$  e desvio de 0.020. Para o valor médio da acurácia de teste, a estratégia de superposição com  $\mu = 32$  apresentou valor médio de 0.921 com desvio de 0.024.

Hiperparâmetros				Treino			Teste		
n	$\mu$	b	Estratégia	Iteração1	Média1	Desvio1	Iteração2	Média2	Desvio2
2	8	2	Sequencial	897	0.741	0.054	943	0.759	0.072
		1	Superposição	948	<b>0.806</b>	0.040	933	<b>0.829</b>	0.060
	16	2	Sequencial	931	0.825	0.031	937	0.840	0.056
		1	Superposição	895	<b>0.880</b>	0.015	846	<b>0.889</b>	0.030
	32	2	Sequencial	923	0.883	0.025	936	0.896	0.032
		1	Superposição	887	<b>0.913</b>	0.020	753	<b>0.921</b>	0.024

Tabela 8 – Melhores acurácias de treino e teste para  $n = 2$

### 5.2.2 $n = 4$

A análise dos gráficos da figura 15 permite extrair conclusões análogas às que foram apresentadas para o caso  $n = 4$  do conjunto de dados IRIS na seção anterior.

A tabela 9 exibe os maiores valores das acurácias médias de treino e teste juntamente com a iteração em que foram obtidos. O melhor valor de acurácia média para o treino foi de 0.927 e ocorreu na iteração 868 da estratégia de batch em superposição com  $\mu = 32$  e desvio de 0.013. Para o valor médio da acurácia de teste, a estratégia de superposição com  $\mu = 32$  apresentou valor médio de 0.938 com desvio de 0.027. É possível constatar ainda que, comparativamente aos valores da tabela 8, as estratégias de batch sequencial e em superposição tiveram uma melhora nos valores de acurácia com o aumento da quantidade de padrões no batch.

### 5.2.3 $n = 8$

A análise dos gráficos da figura 16 permite extrair conclusões análogas às que foram apresentadas para o caso  $n = 8$  do conjunto de dados IRIS na seção anterior.

A tabela 10 exibe os maiores valores das acurácias médias de treino e teste juntamente com a iteração em que foram obtidos. O melhor valor de acurácia média para o treino foi de 0.931 e ocorreu na iteração 828 da estratégia de batch duplo com  $\mu = 32$  e desvio de 0.020.

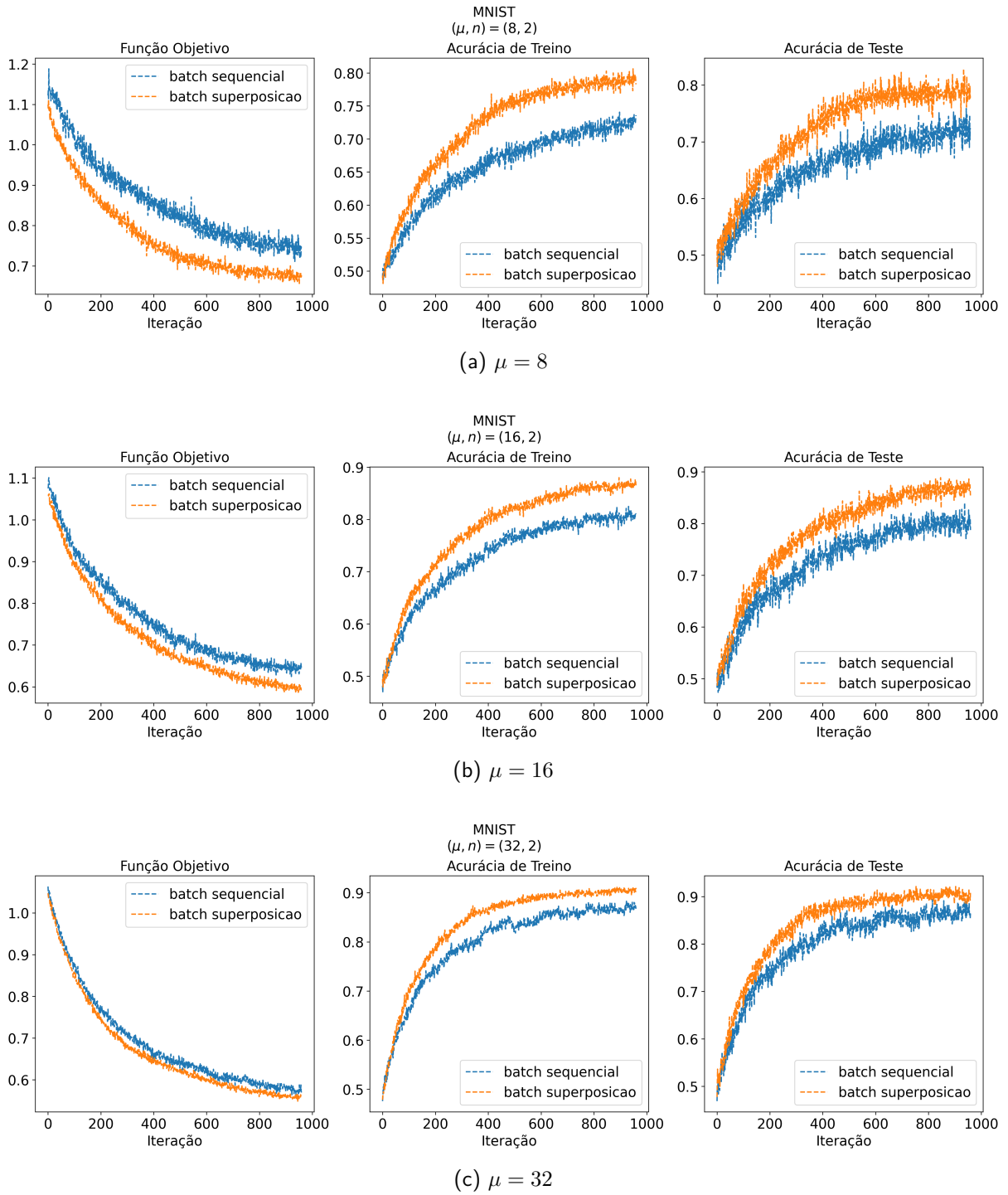


Figura 14 – Resultados MNIST para  $n = 2$

Para o valor médio da acurácia de teste, a estratégia de superposição com  $\mu = 32$  apresentou valor médio de 0.938 com desvio de 0.038. Diferentemente das configurações experimentais anteriores, o caso  $n = 8$  do conjunto de dados MNIST lança a perspectiva de que a estratégia de batch duplo pode apresentar aspectos vantajosos em termos de acurácias frente às demais estratégias.

Hiperparâmetros				Treino			Teste		
n	$\mu$	b	Estratégia	Iteração1	Média1	Desvio1	Iteração2	Média2	Desvio2
4	8	4	Sequencial	847	0.763	0.045	781	0.778	0.045
		1	Superposição	957	<b>0.877</b>	0.008	881	<b>0.886</b>	0.039
		2	Duplo	935	0.837	0.034	929	0.846	0.026
	16	4	Sequencial	798	0.842	0.035	811	0.861	0.053
		1	Superposição	903	<b>0.913</b>	0.017	874	<b>0.920</b>	0.027
		2	Duplo	852	0.888	0.035	823	0.905	0.038
	32	4	Sequencial	917	0.896	0.014	890	0.909	0.043
		1	Superposição	868	<b>0.927</b>	0.013	829	<b>0.938</b>	0.027
		2	Duplo	933	0.917	0.021	953	0.931	0.023

Tabela 9 – Melhores acurácias de treino e teste para  $n = 4$ 

Hiperparâmetros				Treino			Teste		
n	$\mu$	b	Estratégia	Iteração1	Média1	Desvio1	Iteração2	Média2	Desvio2
8	8	8	Sequencial	843	0.788	0.027	930	0.802	0.047
		1	Superposição	923	<b>0.910</b>	0.022	833	<b>0.929</b>	0.029
		2	Duplo	856	0.886	0.027	936	0.892	0.021
		4	Duplo	957	0.837	0.032	878	0.844	0.058
	16	8	Sequencial	934	0.860	0.021	862	0.866	0.038
		1	Superposição	940	<b>0.924</b>	0.021	898	<b>0.930</b>	0.030
		2	Duplo	892	0.906	0.018	734	0.910	0.030
		4	Duplo	946	0.886	0.030	955	0.903	0.045
	32	8	Sequencial	831	0.898	0.021	846	0.914	0.031
		1	Superposição	917	0.928	0.019	505	<b>0.938</b>	0.038
		2	Duplo	828	<b>0.931</b>	0.020	788	0.936	0.024
		4	Duplo	752	0.921	0.017	545	0.928	0.029

Tabela 10 – Melhores acurácias de treino e teste para  $n = 8$ 

### 5.2.4 Síntese dos Resultados

Os resultados para a base MNIST detalhados nas subseções 5.2.1, 5.2.2 e 5.2.3 permitem extrair conclusões similares às que foram discutidas na seção 5.1 para a base IRIS sob a ótica de um novo fator de escala.

No cenário aqui apresentado, houve um aumento da quantidade de amostras por um fator multiplicativo de 4 e da quantidade de qubits por um fator de 3. Mesmo em uma escala maior, fixados os hiperparâmetros  $(n, \mu)$  do batch, ritmos de convergência mais acelerados e acurácias de treino e teste mais altas são obtidos para valores menores de  $b$ .

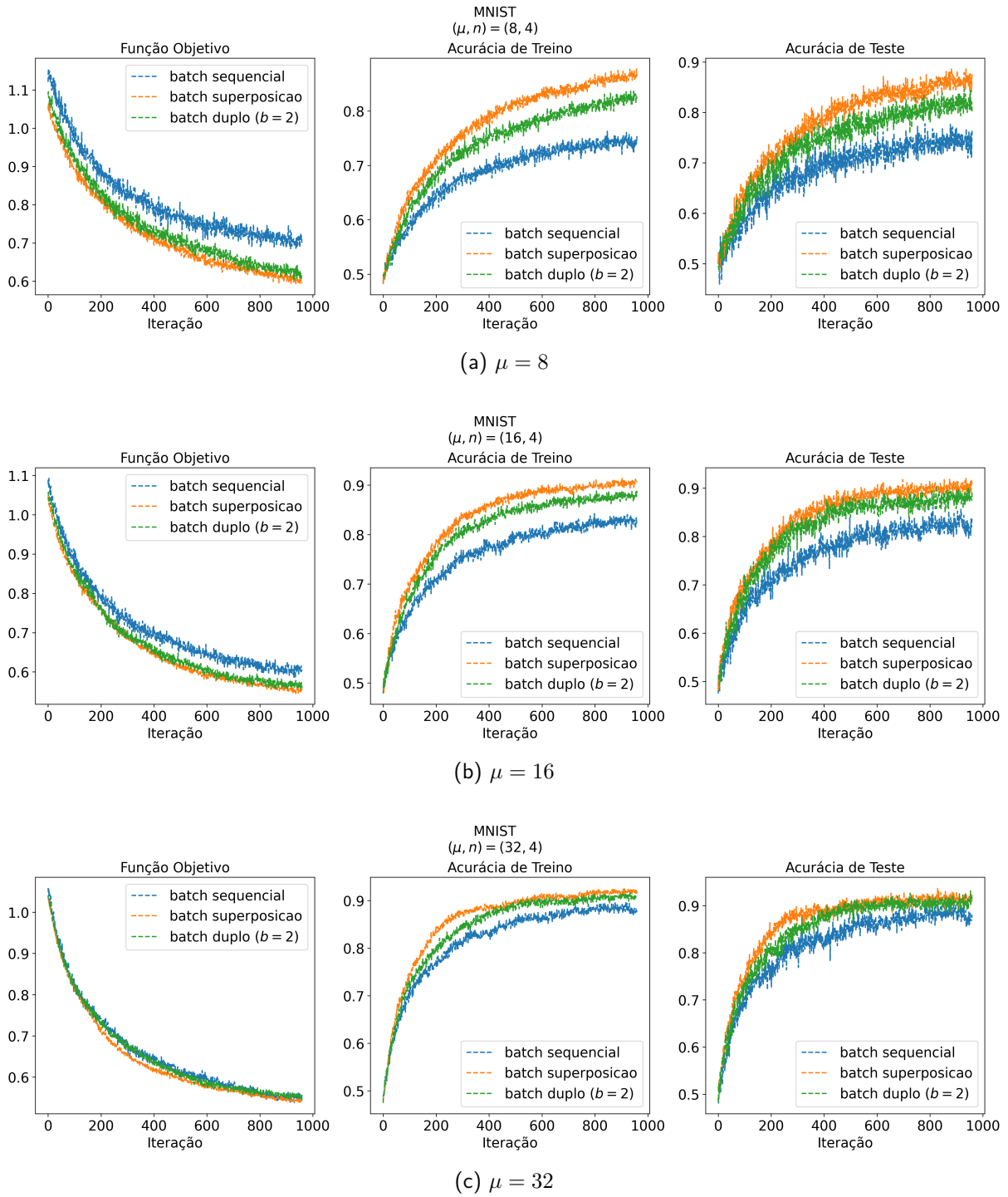
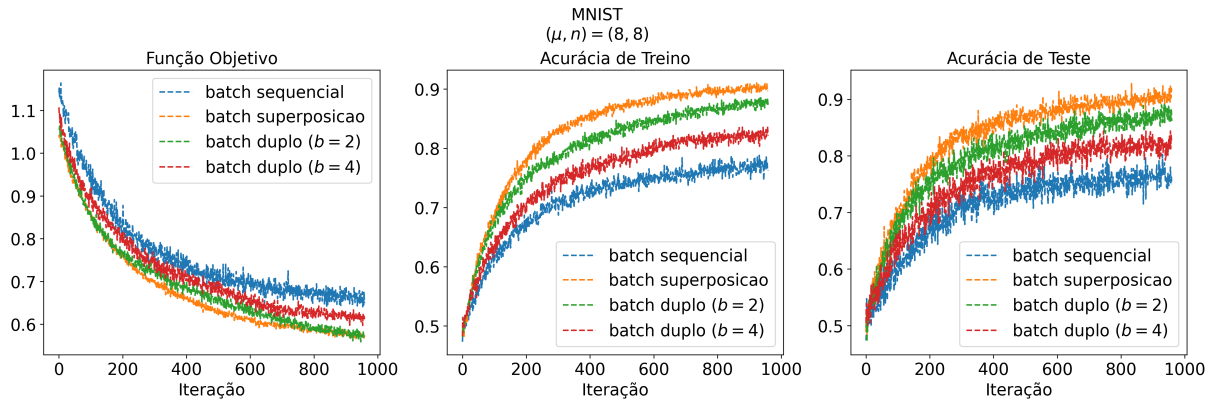
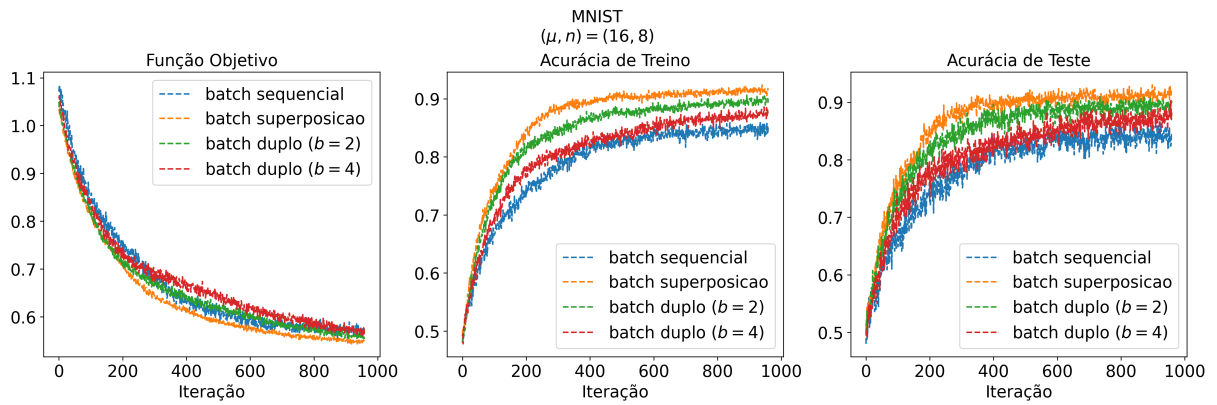
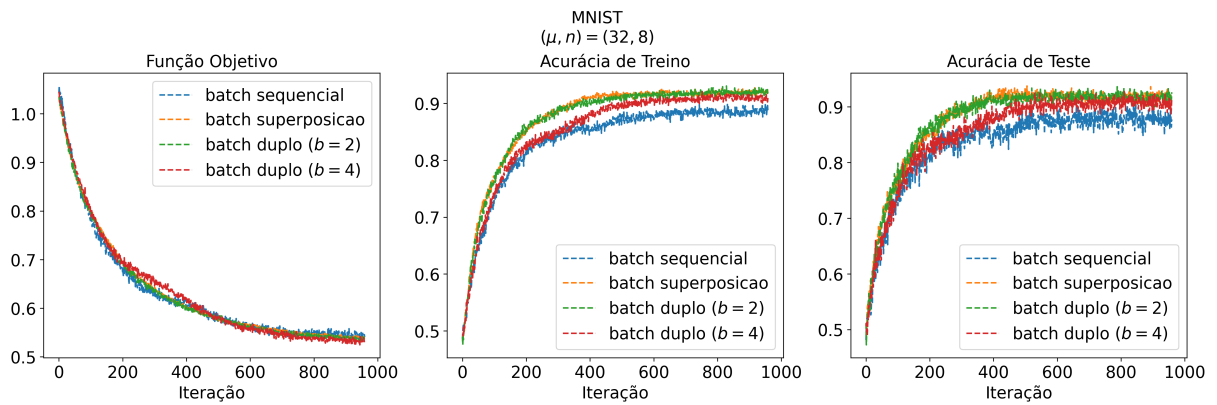


Figura 15 – Resultados MNIST para  $n = 4$

Além do fator de escala, outro diferencial observado na base MNIST em relação à IRIS foi o desempenho das estratégias de batch duplo e de superposição. Para a configuração hiperparamétrica  $(n, \mu) = (8, 32)$ , o maior valor médio da acurácia de treino foi obtido para estratégia de batch duplo com  $b = 2$ . Ainda que pouco expressiva em termos numéricos, essa diferença sugere que possa haver uma ligeira sensibilidade no desempenho das estratégias relacionada ao fator de escala ou a própria natureza dos dados (vide tabela 10).

(a)  $\mu = 8$ (b)  $\mu = 16$ (c)  $\mu = 32$ Figura 16 – Resultados MNIST para  $n = 8$

## 6 CONCLUSÃO

As técnicas de construção de batch da seção 3.2 apresentaram diferentes curvas de desempenho quando valoradas pelo critério definido na subseção 4.2.5. De forma geral, as estratégias com maiores quantidades de padrões em superposição trouxeram resultados de acurácias melhores nos cenários de baixa densidade de shots analisados ( $\mu \in \{8, 16\}$ ). Para  $\mu = 32$ , as estratégias com algum nível de superposição de padrões se mostraram igualmente competitivas com curvas de acurácias praticamente sobrepostas.

O presente estudo também evidencia a importância de hiperparâmetros como a densidade de shots ( $\mu$ ), a quantidade de padrões ( $n$ ) e o nível de superposição ( $b$ ) no batch. Os experimentos com os conjuntos de dados IRIS e MNIST sinalizaram que, mesmo com uma densidade baixa, é possível obter resultados satisfatórios e competitivos.

Do ponto de vista prático, as estratégias que fazem uso de superposição apresentam a desvantagem do uso de qubits adicionais pelo método de construção aqui proposto. Apesar do custo adicional em espaço, essas estratégias se mostraram mais céleres no que tange à convergência da função objetivo.

Possíveis direções de trabalhos futuros emergem das lacunas ainda inexploradas nesse estudo. Uma delas é referente à composição do batch. Até o presente momento, todo o aparato experimental fez uso de batches com padrões de uma única classe. Generalizar a composição do batch pode revelar outras disposições arquiteturais com melhor potencial de empregabilidade.

Outro caminho de investigação está relacionado às expressões analíticas para associar os hiperparâmetros analisados. A análise numérica aponta diferenças de comportamentos para composições  $(n, \mu, b)$  em cada estratégia mas não delimita fronteiras que disciplinem a escolha dos hiperparâmetros ótimos para um determinado contexto. Obter expressões analíticas pode ajudar a delimitar essas fronteiras e aprimorar as técnicas de batch.

## REFERÊNCIAS

- ALZUBI, J.; NAYYAR, A.; KUMAR, A. Machine learning from theory to algorithms: an overview. In: IOP PUBLISHING. *Journal of physics: conference series*. [S.l.], 2018. v. 1142, p. 012012.
- ARAUJO, I. F.; PARK, D. K.; PETRUCCIONE, F.; SILVA, A. J. da. A divide-and-conquer algorithm for quantum state preparation. *Scientific reports*, Nature Publishing Group UK London, v. 11, n. 1, p. 6329, 2021.
- BENEDETTI, M.; LLOYD, E.; SACK, S.; FIORENTINI, M. Parameterized quantum circuits as machine learning models. *Quantum Science and Technology*, IOP Publishing, v. 4, n. 4, p. 043001, 2019.
- BERGHOLM, V.; IZAAC, J.; SCHULD, M.; GOGOLIN, C.; AHMED, S.; AJITH, V.; ALAM, M. S.; ALONSO-LINAJE, G.; AKASHNARAYANAN, B.; ASADI, A. et al. PennyLane: Automatic differentiation of hybrid quantum-classical computations. *arXiv preprint arXiv:1811.04968*, 2018.
- BHARTI, K.; CERVERA-LIERTA, A.; KYAW, T. H.; HAUG, T.; ALPERIN-LEA, S.; ANAND, A.; DEGROOTE, M.; HEIMONEN, H.; KOTTMANN, J. S.; MENKE, T. et al. Noisy intermediate-scale quantum algorithms. *Reviews of Modern Physics*, APS, v. 94, n. 1, p. 015004, 2022.
- BIAMONTE, J.; WITTEK, P.; PANCOTTI, N.; REBENTROST, P.; WIEBE, N.; LLOYD, S. Quantum machine learning. *Nature*, Nature Publishing Group UK London, v. 549, n. 7671, p. 195–202, 2017.
- BOTTOU, L.; CURTIS, F. E.; NOCEDAL, J. Optimization methods for large-scale machine learning. *SIAM review*, SIAM, v. 60, n. 2, p. 223–311, 2018.
- BOTTOU, L. et al. Stochastic gradient learning in neural networks. *Proceedings of Neuro-Nimes*, Nimes, v. 91, n. 8, p. 12, 1991.
- CEREZO, M.; ARRASMITH, A.; BABBUS, R.; BENJAMIN, S. C.; ENDO, S.; FUJII, K.; MCCLEAN, J. R.; MITARAI, K.; YUAN, X.; CINCIO, L. et al. Variational quantum algorithms. *Nature Reviews Physics*, Nature Publishing Group UK London, v. 3, n. 9, p. 625–644, 2021.
- CHEN, S. Y.-C.; HUANG, C.-M.; HSING, C.-W.; KAO, Y.-J. Hybrid quantum-classical classifier based on tensor network and variational quantum circuit. *arXiv preprint arXiv:2011.14651*, 2020.
- CONG, I.; CHOI, S.; LUKIN, M. D. Quantum convolutional neural networks. *Nature Physics*, Nature Publishing Group UK London, v. 15, n. 12, p. 1273–1278, 2019.
- DIEDERIK, K. Adam: A method for stochastic optimization. *(No Title)*, 2014.
- DUAN, B.; SUN, X.; HSIEH, C.-Y. Parallelized variational quantum classifier with shallow qram circuit. *Quantum Information Processing*, Springer, v. 23, n. 3, p. 92, 2024.
- FARHI, E.; NEVEN, H. Classification with quantum neural networks on near term processors. *arXiv preprint arXiv:1802.06002*, 2018.

- FISHER, R. A. The use of multiple measurements in taxonomic problems. *Annals of Eugenics*, Wiley, v. 7, n. 2, p. 179–188, 1936.
- FOUCAULT, M. *As palavras e as coisas: uma arqueologia das ciências humanas*. São Paulo: Martins Fontes, 2002. Tradução de Salma Tannus Muchail.
- FRIEDLANDER, M. P.; SCHMIDT, M. Hybrid deterministic-stochastic methods for data fitting. *SIAM Journal on Scientific Computing*, SIAM, v. 34, n. 3, p. A1380–A1405, 2012.
- GOODFELLOW, I.; BENGIO, Y.; COURVILLE, A.; BENGIO, Y. *Deep learning*. [S.l.]: MIT press Cambridge, 2016. v. 1.
- HARROW, A. W.; NAPP, J. C. Low-depth gradient measurements can improve convergence in variational hybrid quantum-classical algorithms. *Physical Review Letters*, APS, v. 126, n. 14, p. 140502, 2021.
- HAYKIN, S. *Neural Networks and Learning Machines*. 3rd. ed. Upper Saddle River, NJ, USA: Pearson Education, 2008. ISBN 9780131471399.
- HE, F.; LIU, T.; TAO, D. Control batch size and learning rate to generalize well: Theoretical and empirical evidence. *Advances in neural information processing systems*, v. 32, 2019.
- JERBI, S.; FIDERER, L. J.; NAUTRUP, H. P.; KÜBLER, J. M.; BRIEGEL, H. J.; DUNJKO, V. Quantum machine learning beyond kernel methods. *Nature Communications*, Nature Publishing Group, v. 14, n. 1, p. 1–8, 2023.
- KÜBLER, J. M.; ARRASMITH, A.; CINCIO, L.; COLES, P. J. An adaptive optimizer for measurement-frugal variational algorithms. *Quantum*, Verein zur Förderung des Open Access Publizierens in den Quantenwissenschaften, v. 4, p. 263, 2020.
- LAMICHHANE, P.; RAWAT, D. B. Quantum machine learning: Recent advances, challenges and perspectives. *IEEE Access*, IEEE, 2025.
- LAROSE, R.; COYLE, B. Robust data encodings for quantum classifiers. *Physical Review A*, APS, v. 102, n. 3, p. 032420, 2020.
- LECUN, Y.; BOTTOU, L.; BENGIO, Y.; HAFFNER, P. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, IEEE, v. 86, n. 11, p. 2278–2324, 1998.
- LOCKWOOD, O. An empirical review of optimization techniques for quantum variational circuits. *arXiv preprint arXiv:2202.01389*, 2022.
- MAHESHWARI, D.; SIERRA-SOSA, D.; GARCIA-ZAPIRAIN, B. Variational quantum classifier for binary classification: Real vs synthetic dataset. *IEEE access*, IEEE, v. 10, p. 3705–3715, 2021.
- MITCHELL, T. M. *Machine Learning*. 1st. ed. New York, NY, USA: McGraw-Hill, Inc., 1997.
- MONTANARO, A. Quantum algorithms: an overview. *npj Quantum Information*, Nature Publishing Group, v. 2, n. 1, p. 1–8, 2016.
- MOTTONEN, M.; VARTIAINEN, J. J.; BERGHOLM, V.; SALOMAA, M. M. Transformation of quantum states using uniformly controlled rotations. *arXiv preprint quant-ph/0407010*, 2004.



- NIELSEN, M. A.; CHUANG, I. L. *Quantum computation and quantum information*. [S.l.]: Cambridge university press, 2010.
- PEDREGOSA, F.; VAROQUAUX, G.; GRAMFORT, A.; MICHEL, V.; THIRION, B.; GRISEL, O.; BLONDEL, M.; PRETTENHOFER, P.; WEISS, R.; DUBOURG, V.; VANDERPLAS, J.; PASSOS, A.; COURNAPEAU, D.; BRUCHER, M.; PERROT, M.; DUCHESNAY, É. Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, v. 12, p. 2825–2830, 2011.
- PIRA, L.; FERRIE, C. On the interpretability of quantum neural networks. *Quantum Machine Intelligence*, Springer, v. 6, n. 2, p. 52, 2024.
- QI, H.; XIAO, S.; LIU, Z.; GONG, C.; GANI, A. Variational quantum algorithms: fundamental concepts, applications and challenges. *Quantum Information Processing*, Springer, v. 23, n. 6, p. 224, 2024.
- ROMERO, J.; OLSON, J. P.; ASPURU-GUZZI, A. Quantum autoencoders for efficient compression of quantum data. *Quantum Science and Technology*, IOP Publishing, v. 2, n. 4, p. 045001, 2017.
- RUDER, S. An overview of gradient descent optimization algorithms. *arXiv preprint arXiv:1609.04747*, 2016.
- SCHULD, M. Supervised quantum machine learning models are kernel methods. *arXiv preprint arXiv:2101.11020*, 2021.
- SCHULD, M.; BERGHOLM, V.; GOGOLIN, C.; IZAAC, J.; KILLORAN, N. Evaluating analytic gradients on quantum hardware. *Physical Review A*, APS, v. 99, n. 3, p. 032331, 2019.
- SCHULD, M.; BOCHAROV, A.; SVORE, K. M.; WIEBE, N. Circuit-centric quantum classifiers. *Physical Review A*, APS, v. 101, n. 3, p. 032308, 2020.
- SCHULD, M.; PETRUCCIONE, F. *Machine learning with quantum computers*. [S.l.]: Springer, 2021. v. 676.
- SCHULD, M.; SINAYSKIY, I.; PETRUCCIONE, F. An introduction to quantum machine learning. *Contemporary Physics*, Taylor & Francis, v. 56, n. 2, p. 172–185, 2015.
- SCHULD, M.; SWEKE, R.; MEYER, J. J. Effect of data encoding on the expressive power of variational quantum-machine-learning models. *Physical Review A*, APS, v. 103, n. 3, p. 032430, 2021.
- SIM, S.; JOHNSON, P. D.; ASPURU-GUZZI, A. Expressibility and entangling capability of parameterized quantum circuits for hybrid quantum-classical algorithms. *Advanced Quantum Technologies*, Wiley Online Library, v. 2, n. 12, p. 1900070, 2019.
- SWEKE, R.; WILDE, F.; MEYER, J.; SCHULD, M.; FÄHRMANN, P. K.; MEYNARD-PIGANEAU, B.; EISERT, J. Stochastic gradient descent for hybrid quantum-classical optimization. *Quantum*, Verein zur Förderung des Open Access Publizierens in den Quantenwissenschaften, v. 4, p. 314, 2020.
- TEAM, K. *MNIST handwritten digits dataset - Keras*. 2024. <<https://keras.io/api/datasets/mnist/>>. Accessed: 2025-05-27.