

UNIVERSIDADE FEDERAL DE ITAJUBÁ
PROGRAMA DE PÓS-GRADUAÇÃO EM
ENGENHARIA DE PRODUÇÃO

**Utilização de Aprendizagem de Máquina para a Inclusão de Fatores
Emocionais Humanos em projetos de Simulação**

Yago Toledo Lima

Itajubá, maio de 2023

UNIVERSIDADE FEDERAL DE ITAJUBÁ
PROGRAMA DE PÓS-GRADUAÇÃO EM
ENGENHARIA DE PRODUÇÃO

Yago Toledo Lima

**Utilização de Aprendizagem de Máquina para a Inclusão de Fatores
Emocionais Humanos em projetos de Simulação**

**Dissertação submetida ao programa de Pós-
Graduação em Engenharia de Produção como
parte dos requisitos para obtenção do Título de
Mestre em Ciências em Engenharia de Produção.**

Área: Engenharia de Produção

Orientador: Prof. Dr. José Arnaldo Barra
Montevecchi

Mai de 2023

Itajubá

DEDICATÓRIA

À minha mãe, Maria Tereza, ao meu pai, Neire Márcio e meus irmãos, Cintia e Hadriel por todo o apoio, ensinamentos e paciência que fizeram de mim quem sou.

AGRADECIMENTOS

Aos meus pais, Maria Tereza Brandão Toledo Lima e Neire Marcio Lima, pelos ensinamentos, incentivo, apoio e pela minha formação.

Aos meus irmãos, Cintia e Hadriel pelo apoio e pelos conselhos.

À minha namorada, Weena, pelo apoio e consolo nos momentos mais difíceis.

Ao meu orientador, José Arnaldo Barra Montevechi, pela orientação deste trabalho, pelos ensinamentos e pela paciência, principalmente diante das minhas falhas e teimosia.

Aos professores e servidores da UNIFEI, pela dedicação em oferecer a educação de qualidade que a universidade fornece.

A todos colegas do NEAAD, em especial João Vítor, Carlos, Gustavo, Alyson, Wesley e Afonso pela ajuda, conselhos e pela amizade.

Ao IEPG UNIFEI pela oportunidade que me foi concedida.

A CAPES, FAPEMIG e CNPq pelo apoio e incentivo à pesquisa brasileira e pelos apoios fornecidos à Universidade Federal de Itajubá que possibilitou esta pesquisa.

A todos vocês, meu muito obrigado!

EPÍGRAFE

“Estude muito aquilo que mais te interessa da maneira mais indisciplinada, irreverente e original possível.”

Richard Feynman

RESUMO

A simulação computacional é uma ferramenta amplamente utilizada para monitorar e otimizar indicadores de desempenho tanto na academia quanto na indústria. Os operadores humanos, no entanto, são comumente representados como recursos de produção nominal constante nos modelos de simulação. Tal modelo, em muitos casos, é insuficiente, uma vez que desconsidera as variações inerentes da fisionomia humana. Há na literatura formas de se representar tais variações no desempenho humano, porém, o fator emocional raramente é abordado nestes trabalhos. Além disso, os trabalhos que buscam modelar a atitude psicofísica do operador, que engloba as emoções, no geral, necessitam de equipamentos sofisticados ou formulários morosos a serem preenchidos. Para apresentar uma alternativa para a inclusão do fator emocional humano em modelos de simulação, este trabalho utiliza um modelo de aprendizagem de máquina para identificar o humor aparente de operadores a partir de filmagens de uma linha de produção de manufatura em operação. Esta abordagem não interfere na carga de trabalho dos operadores nem demanda nenhum tipo de equipamento extra além de uma câmera filmadora comum, preenchendo então a lacuna observada na literatura. Este trabalho demonstrou que há 66 correlações estatísticas significantes entre as variáveis de humor detectadas pelo modelo de aprendizagem de máquina utilizado e o tempo de operação na linha. Também foi demonstrado que as curvas de distribuição de probabilidade para o tempo de operação variam significativamente quando consideradas diferentes classes de humor. Por fim, foi demonstrada a aplicação destas curvas em um modelo de simulação. Os resultados evidenciam uma tendência de melhoria nos resultados do modelo, demonstrando assim a viabilidade da técnica aplicada. Além dos resultados demonstrados, este trabalho também apresenta instruções para aplicação da técnica proposta que podem ser utilizadas para projetos similares. Além das simulações, a técnica apresentada pode também ser utilizada em diversas áreas incluindo prevenção de defeitos de fabricação, saúde e segurança ocupacional.

Palavras-chave: Simulação, Desempenho humano, Fator emocional, Aprendizagem de máquina.

ABSTRACT

Computational simulation is a widely used tool for monitoring and optimizing performance indicators in both academia and industry. However, human operators are commonly represented as resources with constant nominal production in simulation models. Such models are often insufficient, as they disregard the inherent variations due to human physiology. Although there are ways to represent such variations in human performance in the academic literature, emotional factors are rarely addressed. Moreover, the studies that seek to model the psychophysical attitude, which includes emotions, of the operator usually require sophisticated equipment or time-consuming forms to be filled by the operators. To present an alternative for the inclusion of the emotional human factor in simulation models, this work uses a machine learning model to identify the apparent mood of operators from footage of a manufacturing production line in operation. This approach does not interfere with the workload of the operators nor require any extra equipment other than a common video camera, filling a gap observed in the literature. This study revealed 66 statistically significant correlations between the mood variables estimated by the machine learning model and operation time on the assembly line. It was also shown that the probability distribution functions for the operation time differ significantly when considering different classes of mood. Lastly, those different curves were demonstrated in a simulation model. The results show an improvement trend in the model's results, thus demonstrating the viability of this technique. This study also provides instructions for applying the proposed technique that can be used in similar projects. Besides simulations, this technique can be applied to a variety of fields including defects prevention, occupational health and safety.

Keywords: *Simulation, Human performance, Emotional factor, Machine learning.*

LISTA DE ILUSTRAÇÕES

Figura 1 – Classificação da pesquisa	17
Figura 2 – Etapas da pesquisa	19
Figura 3 – Etapas de um projeto de simulação a eventos discretos	22
Figura 4 – Um sistema típico de aprendizagem de máquina	26
Figura 5 – Taxonomia das técnicas de aprendizagem de máquina	27
Figura 6 – Árvore de decisão simples	29
Figura 7 – Uma rede neural simples	30
Figura 8 – Modelo básico de Aprendizado por Reforço	32
Figura 9 – Comparação de imagens com e sem contexto	33
Figura 10 – Modelo EMOTIC	35
Figura 11 – Diagrama de fluxo de dados	43
Figura 12 – Linha de montagem de <i>Karts Lego</i> [®]	45
Figura 13 – IDEF-SIM da linha de montagem de <i>Karts Lego</i> [®] página 1 de 3	46
Figura 14 – IDEF-SIM da linha de montagem de <i>Karts Lego</i> [®] página 2 de 3	47
Figura 15 – IDEF-SIM da linha de montagem de <i>Karts Lego</i> [®] página 3 de 3	48
Figura 16 – <i>Software</i> adaptado	49
Figura 17 – Pessoas detectadas	52
Figura 18 – Ponto central	53
Figura 19 – Validação do ponto central	54
Figura 20 – <i>Boxplot</i> de duração do processo por classe para o operador 1	61
Figura 21 – <i>Boxplot</i> de duração do processo por classe para o operador 2	62
Figura 22 – <i>Boxplot</i> de duração do processo por classe para o operador 3	62
Figura 23 – <i>Boxplot</i> de duração do processo por classe para o operador 4	63
Figura 24 – <i>Boxplot</i> de duração do processo por classe para o operador 5	63
Figura 25 – Histogramas para o operador 1	67
Figura 26 – Histogramas para o operador 2	67
Figura 27 – Histogramas para o operador 3	68
Figura 28 – Histogramas para o operador 4	68
Figura 29 – Histogramas para o operador 5	69

Figura 30 – Modelo de simulação	70
Figura 31 – <i>Global Table</i> de distribuições	73
Figura 32 – <i>Boxplots</i> dos resultados da simulação	75
Figura 33 – Histograma da classe base para o operador 1	96
Figura 34 – Histograma da classe para o operador 1 de 0% a 25%	96
Figura 35 – Histograma da classe para o operador 1 de 25% a 50%	97
Figura 36 – Histograma da classe para o operador 1 de 50% a 75%	97
Figura 37 – Histograma da classe para o operador 1 de 75% a 100%	98
Figura 38 – Histograma da classe base para o operador 2	98
Figura 39 – Histograma da classe para o operador 2 de 0% a 25%	99
Figura 40 – Histograma da classe para o operador 2 de 25% a 50%	99
Figura 41 – Histograma da classe para o operador 2 de 50% a 75%	100
Figura 42 – Histograma da classe para o operador 2 de 75% a 100%	100
Figura 43 – Histograma da classe base para o operador 3	101
Figura 44 – Histograma da classe para o operador 3 de 0% a 25%	101
Figura 45 – Histograma da classe para o operador 3 de 25% a 50%	102
Figura 46 – Histograma da classe para o operador 3 de 50% a 75%	102
Figura 47 – Histograma da classe para o operador 3 de 75% a 100%	103
Figura 48 – Histograma da classe base para o operador 4	103
Figura 49 – Histograma da classe para o operador 4 de 0% a 25%	104
Figura 50 – Histograma da classe para o operador 4 de 25% a 50%	104
Figura 51 – Histograma da classe para o operador 4 de 50% a 75%	105
Figura 52 – Histograma da classe para o operador 4 de 75% a 100%	105
Figura 53 – Histograma da classe base para o operador 5	106
Figura 54 – Histograma da classe para o operador 5 de 0% a 25%	106
Figura 55 – Histograma da classe para o operador 5 de 25% a 50%	107
Figura 56 – Histograma da classe para o operador 5 de 50% a 75%	107
Figura 57 – Histograma da classe para o operador 5 de 75% a 100%	108

LISTA DE TABELAS

Tabela 1 – Variáveis do EMOTIC.	33
Tabela 2 – Resultados da RSL	38
Tabela 3 – Trechos desconsiderados na cronoanálise.	50
Tabela 4 – Totais da cronoanálise.	51
Tabela 5 – Correlação da emoção aparente com a duração do processo.	57
Tabela 6 – Variáveis selecionadas para cada operador.	59
Tabela 7 – Classes de humor por operador.	60
Tabela 8 – Classes de humor por operador.	64
Tabela 9 – Cenário Proposto.	71
Tabela 10 – Valores médios extraídos da simulação.	74
Tabela 11 – Correlação da emoção aparente com a duração do processo.	89
Tabela 12 – Funções para o tempo de operação no Flexsim	94
Tabela 13 – Simbologia do IDEF-SIM	110

LISTA DE ABREVIATURAS E SIGLAS

ERP	<i>Enterprise Resource Planning</i>
IDEF-SIM	<i>Integrated Definition Methods - Simulation</i>
k-NN	<i>k-Nearest Neighbor</i>
RSL	Revisão Sistemática de Literatura
SED	Simulação a Eventos Discretos

SUMÁRIO

1	INTRODUÇÃO	14
1.1	Objetivos	15
1.2	Método de pesquisa	16
1.2.1	Classificação da pesquisa	16
1.2.2	Etapas da pesquisa	18
1.3	Estrutura do trabalho	20
2	FUNDAMENTAÇÃO TEÓRICA	21
2.1	Modelagem e Simulação	21
2.1.1	O fator humano em simulações	23
2.2	Aprendizagem de Máquina	25
2.2.1	Aprendizado Supervisionado	27
2.2.2	Aprendizado não supervisionado	31
2.2.3	Aprendizado por reforço	31
2.3	Detecção de Humor	32
2.3.1	O modelo EMOTIC	32
3	REVISÃO SISTEMÁTICA DA LITERATURA	36
3.1	Método de Revisão Sistemática da Literatura	36
3.1.1	Projeto	37
3.1.2	Busca e Seleção	37
3.1.3	Análise e Síntese	38
3.1.4	Apresentação dos Resultados	38
3.2	Resultados e discussões	38
3.2.1	Fatores Humanos Considerados	40
3.2.2	Técnicas para modelar o desempenho humano	40
3.2.3	Lacunas na Literatura	41
4	DESENVOLVIMENTO E DISCUSSÕES	43
4.1	Preparação dos dados	44
4.1.1	Obtenção das gravações	44

4.1.2	Definição da taxa e extração dos quadros	48
4.1.3	Cronoanálise dos quadros	49
4.1.4	Detecção dos indivíduos	51
4.1.5	Identificação dos operadores	52
4.2	Identificação das emoções	54
4.2.1	Classificação do humor aparente	54
4.2.2	Cruzamento de dados	55
4.2.2.1	Identificação das iterações dos processos e cálculo da duração	55
4.2.2.2	Cálculo da intensidade das variáveis de humor aparente dentro da iteração do processo	56
4.3	Elaboração do modelo	56
4.3.1	Correlação do humor com o tempo de processamento	56
4.3.2	Seleção de variáveis	58
4.3.3	Definição das classes de humor	59
4.3.4	Cálculo das curvas de distribuição de probabilidade	64
4.4	Validação do modelo	69
4.4.1	Preparação dos cenários	70
4.4.2	Atualização do modelo de simulação	73
4.4.3	Execução do modelo e comparação dos resultados	74
5	CONSIDERAÇÕES FINAIS	77
5.1	Sugestões para trabalhos futuros	80
	 REFERÊNCIAS	 82
	 APÊNDICES	 88
	APÊNDICE A – CORRELAÇÕES	89
	APÊNDICE B – FUNÇÕES PARA O TEMPO DE OPERAÇÃO	94
	APÊNDICE C – HISTOGRAMAS	96

	ANEXOS	109
	ANEXO A – SIMBOLOGIA DO IDEF-SIM	110
	ANEXO B – COMANDOS FFMPEG	111
	ANEXO C – <i>SCRIPT</i> PARA DETERMINAR PONTOS CENTRAIS .	112
	ANEXO D – <i>SCRIPT</i> PARA CRUZAMENTO DE DADOS	114
	ANEXO E – <i>SCRIPT</i> PARA VALIDAÇÃO DO PONTO CENTRAL . .	124
E.1	Arquivo <i>draw_operator_samples.py</i>	124
E.2	Arquivo <i>utils.py</i>	125
E.3	Arquivo <i>requirements.txt</i>	129
E.4	Comando para executar o <i>script</i>	130
	ANEXO F – <i>SCRIPTS</i> PARA CÁLCULO DA CORRELAÇÃO E SE-	
	LEÇÃO DE VARIÁVEIS	131
F.1	Arquivo <i>common.R</i>	131
F.2	Arquivo <i>correlation.R</i>	137
F.3	Arquivo <i>fit.R</i>	141
	ANEXO G – PUBLICAÇÕES	143

1 INTRODUÇÃO

No contexto da produção industrial existe uma pressão para a melhoria e otimização de processos produtivos. Em muitos casos a redução de desperdícios e melhoria da eficiência em geral podem significar uma vantagem competitiva importante. Modelagem e simulação, dentro da pesquisa operacional, são amplamente utilizadas na indústria e na academia para monitorar e otimizar indicadores de desempenho (ADEWUNMI; AICKELIN; BANGSOW, 2012). Segundo Pecek e Kovacic (2011) as simulações computacionais podem ser utilizadas como ferramenta de análise nos ambientes estocásticos em que os recursos são limitados, características comuns no ambiente de produção industrial.

A importância de se representar o fator humano em modelos de simulação é indiscutível, pois as interações humanas afetam diretamente ou indiretamente os resultados daquilo que se deseja simular no mundo real (OLIVEIRA, 2017). Para Digiesi *et al.* (2009), o comportamento humano e seu desempenho em linhas de produção podem ser afetados por diversos fatores: o ambiente de trabalho (físico: microclima, ergonomia e barulho; social: relações humanas e comunicação com o grupo), a natureza da tarefa (discreta ou contínua e motora ou cognitiva) e fatores pessoais (atitude psicofísica, habilidade pessoal, idade e sexo). No entanto, segundo os autores, em simulações, operadores humanos, em geral, são representados como recursos simples com um desempenho nominal constante no tempo, sem considerar as variações dinâmicas do desempenho humano.

Este trabalho faz uso de um fator pessoal: o humor aparente do operador. Na literatura, é possível encontrar vários trabalhos apontando a relação entre o humor do operador e seu desempenho, como Heimstra, Ellingstad e Dekock (1967), Hirt *et al.* (1996), Miner e Glomb (2010) e Calle *et al.* (2021). No entanto, para mensurar e classificar o humor dos operadores, nos exemplos citados, são utilizados questionários que devem ser respondidos pelos operadores antes ou durante as operações. Tal abordagem torna o processo moroso e dificulta a análise da variação dessas variáveis ao longo do tempo.

Zhang *et al.* (2017) utiliza aprendizagem de máquina para modelar fatores pessoais do desempenho humano sem a necessidade do preenchimento de formulários por parte dos operadores. No entanto, este trabalho fez uso de eletroencefalogramas, dificultando seu uso pela necessidade de um equipamento sofisticado e de difícil acesso.

Esta pesquisa faz uso do modelo EMOTIC, que é um modelo de aprendizagem de máquina treinado para identificar o humor aparente de pessoas em uma imagem (KOSTI *et al.*, 2019). Esse modelo se destaca por analisar além da expressão facial e corporal do indivíduo, o contexto em que ele se encontra na imagem. Uma vez que o modelo EMOTIC é capaz de extrair o humor aparente de operadores a partir de imagens contendo-os, este trabalho apresenta uma alternativa para modelar este fator em modelos de simulação sem que seja necessário onerar o operador com atividades extra nem obter equipamentos sofisticados e de difícil acesso.

Montevechi *et al.* (2010) definem três etapas principais para um projeto de simulação: concepção, implementação e análise. Na etapa de concepção acontece a modelagem dos dados de entrada em que são produzidos tempos, custos, capacidades, entre outros fatores relacionados ao modelo que se deseja simular. Este trabalho propõe a adição do humor aparente do operador dentro destes dados de entrada. É importante salientar que o humor aparente é um dado complementar a outras fontes e não um substituto.

1.1 Objetivos

Devido à importância da representação do fator humano para a fidelidade de modelos de simulação, sabendo da dificuldade de se mensurar fatores pessoais, como o humor do operador, e conhecendo a capacidade da aprendizagem de máquina para criar soluções voltadas ao processamento de dados brutos de sensores, o presente trabalho se propõe a utilizar um modelo de aprendizagem de máquina para detectar o humor dos operadores de uma linha de montagem manufaturada a partir de filmagens desta linha em operação. A partir dos resultados deseja-se identificar quais variáveis possuem correlação estatística com o tempo de operação na linha, ou seja, identificar em quais variáveis uma variação representa uma variação no tempo de operação de maneira previsível e estatisticamente significativa. Por fim, deseja-se demonstrar como esses dados podem ser incluídos em um modelo de simulação e verificar possíveis melhorias no modelo.

Para isso, os seguintes objetivos específicos foram identificados:

- Identificar características do humor do operador a partir das gravações em vídeo do processo produtivo.
- Verificar se há ou não correlação das variáveis de humor do operador com o tempo de

operação.

- Comparar os resultados de um modelo de simulação que leva em consideração estas variáveis aos do mesmo modelo sem levá-las em consideração.

Considerando os objetivos e o contexto da pesquisa, três hipóteses foram levantadas. Estas serão confirmadas ou não ao longo da pesquisa:

1. Um ou mais atributos de humor identificados pelo modelo de aprendizagem de máquina possuem correlação estatística com o tempo de operação.
2. A curva de distribuição de probabilidade dos tempos de processamento é diferente quando as amostras consideram classes diferentes de humor para um operador.
3. Um modelo de simulação cujas curvas de distribuição de probabilidade consideram o humor do operador produz melhores resultados.

1.2 Método de pesquisa

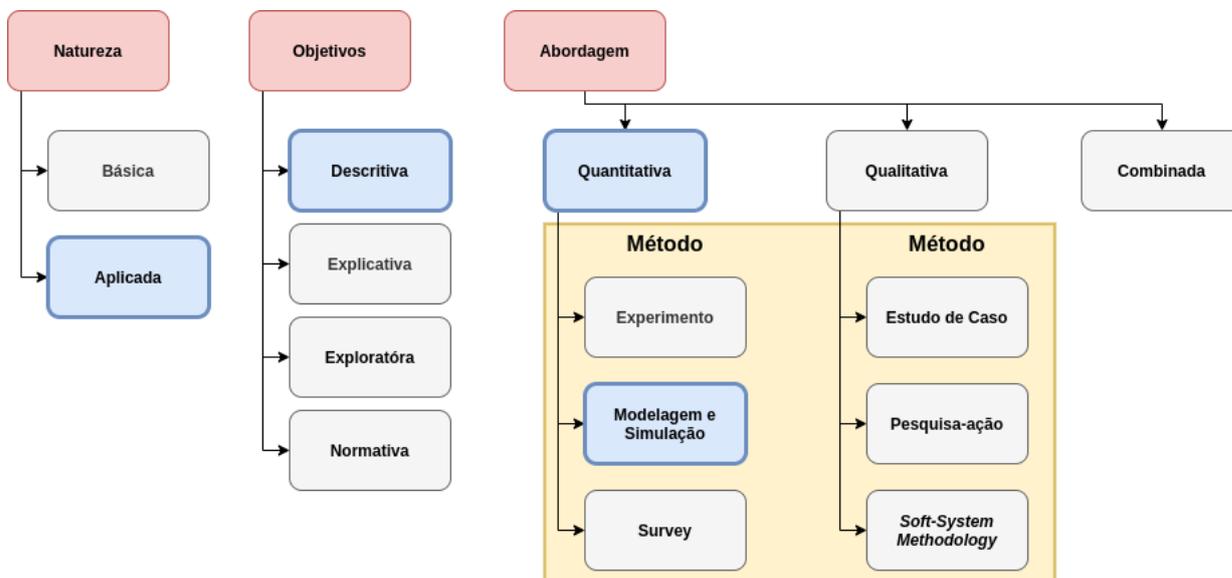
Esta seção apresenta o método de pesquisa utilizado, a justificativa científica para essa pesquisa bem como a classificação do método. Além disso, é apresentada uma visão geral das etapas que foram realizadas durante o desenvolvimento.

1.2.1 Classificação da pesquisa

Miguel *et al.* (2012) classificam pesquisas científicas quanto à natureza, aos objetivos, à abordagem e ao método. Essas encontram-se resumidas na Figura 1 onde estão destacadas as classificações desta pesquisa.

Miguel *et al.* (2012) dividem as pesquisas científicas em duas categorias quanto a sua natureza: básica e aplicada. Uma pesquisa básica é aquela cujo objetivo se restringe ao incremento do conhecimento científico, ao passo que a pesquisa aplicada tem em seus objetivos aplicações comerciais, ou seja, tem em vista o desenvolvimento de novos processos ou produtos voltados para as necessidades do mercado (APPOLINÁRIO, 2012). Esta pesquisa, portanto, se trata de uma pesquisa aplicada, pois é voltada à melhoria de uma ferramenta para tomada de decisão dentro do processo produtivo.

Figura 1 – Classificação da pesquisa



Fonte: adaptado de Miguel *et al.* (2012)

Considerando as quatro categorias definidas por Miguel *et al.* (2012) quanto a seus objetivos, bem como os objetivos gerais e específicos desta pesquisa, ela é considerada descritiva. Uma pesquisa descritiva objetiva criar um modelo que descreva adequadamente as relações causais da realidade (BERTRAND; FRANSOO, 2002).

Miguel *et al.* (2012) traçam a distinção entre as abordagens quantitativa e qualitativa quanto ao nível de interferência dos indivíduos que estão sendo estudados. Segundo os autores, numa pesquisa quantitativa, as variáveis e métricas são definidas e mensuradas de maneira independente dos indivíduos alvo do estudo, já em abordagens qualitativas a perspectiva desses indivíduos é objeto do estudo.

Nenhuma pesquisa é completamente quantitativa ou qualitativa. Para fins de classificação, pode-se observar a predominância ou não de uma abordagem ou outra (APPOLINÁRIO, 2012). Esta pesquisa, portanto, possui abordagem quantitativa, pois as métricas utilizadas para verificação das hipóteses propostas foram definidas e mensuradas de maneira independente dos indivíduos pesquisados.

Quanto ao método, essa pesquisa é de modelagem e simulação. Para Miguel *et al.* (2012) a modelagem e simulação procura representar um sistema produtivo a partir de modelos matemáticos. No caso dessa pesquisa o objetivo é propor uma melhoria na representação

matemática para operadores humanos em sistemas produtivos.

1.2.2 Etapas da pesquisa

Essa pesquisa foi dividida em quatro etapas principais: preparação, identificação das emoções, elaboração do modelo e validação do modelo. A Figura 2 apresenta os procedimentos de cada etapa de maneira resumida.

Na primeira etapa, os dados serão preparados para que seja possível reconhecer as emoções com o modelo EMOTIC. Para isso vários procedimentos precisarão ser realizados. Primeiro será necessário extrair um número adequado de quadros dos vídeos das filmagens. Um processo iterativo será utilizado para garantir que a variação de um quadro para outro permita uma cronoanálise precisa. Em seguida será feita a cronoanálise quadro a quadro das gravações.

O modelo EMOTIC precisa receber, além da imagem a ser processada, a região onde a pessoa cujas emoções serão detectadas se encontra. Para isso, um segundo modelo de aprendizagem de máquina será utilizado para detectar estes indivíduos. Além disso, o operador de cada posto de trabalho precisará ser identificado.

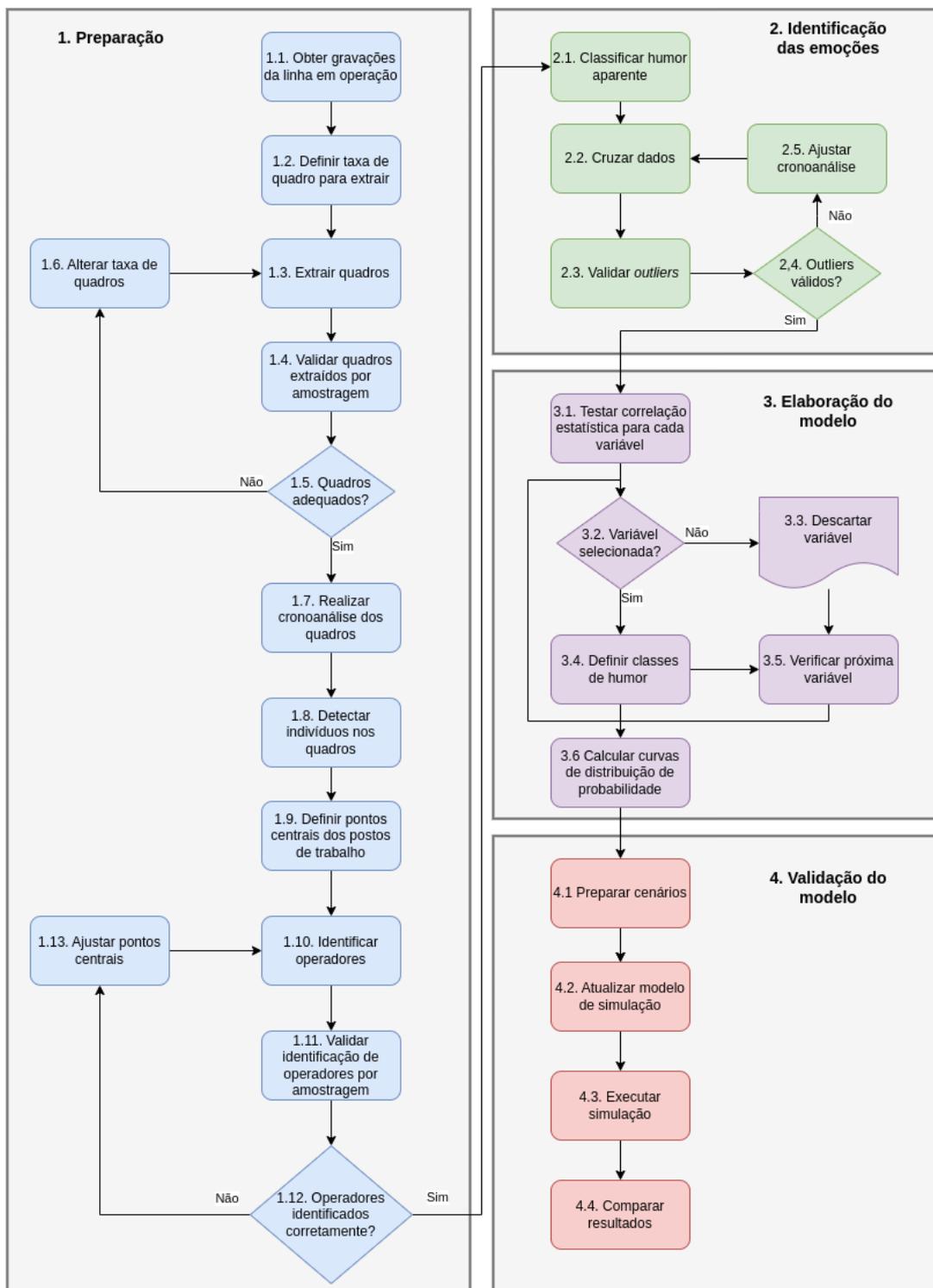
Na segunda etapa, o modelo EMOTIC será utilizado para classificar o humor aparente de cada operador em cada quadro das filmagens. Além disso, será necessário cruzar os dados disponíveis para associar cada classificação de humor a seu respectivo operador a sua respectiva iteração no processo de montagem.

A terceira etapa iniciará-se com a verificação se o humor aparente do operador afeta seu desempenho, ou seja, se há correlação estatística entre as variáveis de humor detectadas pelo modelo de aprendizagem de máquina e o tempo de operação cronometrado. Será nesta etapa que ocorrerá a validação ou não das hipóteses 1 e 2 propostas na seção 1.1:

- Hipótese 1: um ou mais atributos de humor identificados pelo modelo de aprendizagem de máquina possuem correlação estatística com o tempo de operação.
- Hipótese 2: a curva de distribuição de probabilidade dos tempos de processamento é diferente quando as amostras consideram classes diferentes de humor para um operador.

Ainda na terceira etapa, serão selecionadas as variáveis que irão ser representadas no modelo de simulação das etapas seguintes. Será elaborado também um modelo matemático que

Figura 2 – Etapas da pesquisa



Fonte: Autor.

representa os tempos de operação levando em consideração a emoção aparente do operador. Este modelo só será viável caso as hipóteses 1 ou 2 sejam confirmadas. Como resultado dessa etapa, deseja-se obter uma ou mais curvas de distribuição de probabilidade para os tempos que levem em consideração o humor aparente do operador.

Na quarta etapa, o modelo matemático proposto na terceira será testado. Nela, o modelo de simulação desenvolvido por Rocha (2020) será atualizado com o modelo matemático criado na terceira etapa. Os resultados do modelo antes e depois serão então comparados. Assim espera-se confirmar ou não a hipótese 3 dos objetivos deste trabalho.

1.3 Estrutura do trabalho

Após a presente introdução, que conclui o Capítulo 1, o Capítulo 2, traz uma visão geral dos principais conceitos relevantes para este trabalho. Em seguida, no Capítulo 3 é apresentada uma revisão sistemática da literatura sobre a representação do desempenho humano em simulações. No Capítulo 4 é descrita a aplicação do método e a execução do roteiro para alcançar os objetivos propostos. Por fim, no Capítulo 5 são apresentadas as considerações finais desta pesquisa tendo em vista os objetivos, os resultados obtidos e sugestões para trabalhos futuros.

2 FUNDAMENTAÇÃO TEÓRICA

2.1 Modelagem e Simulação

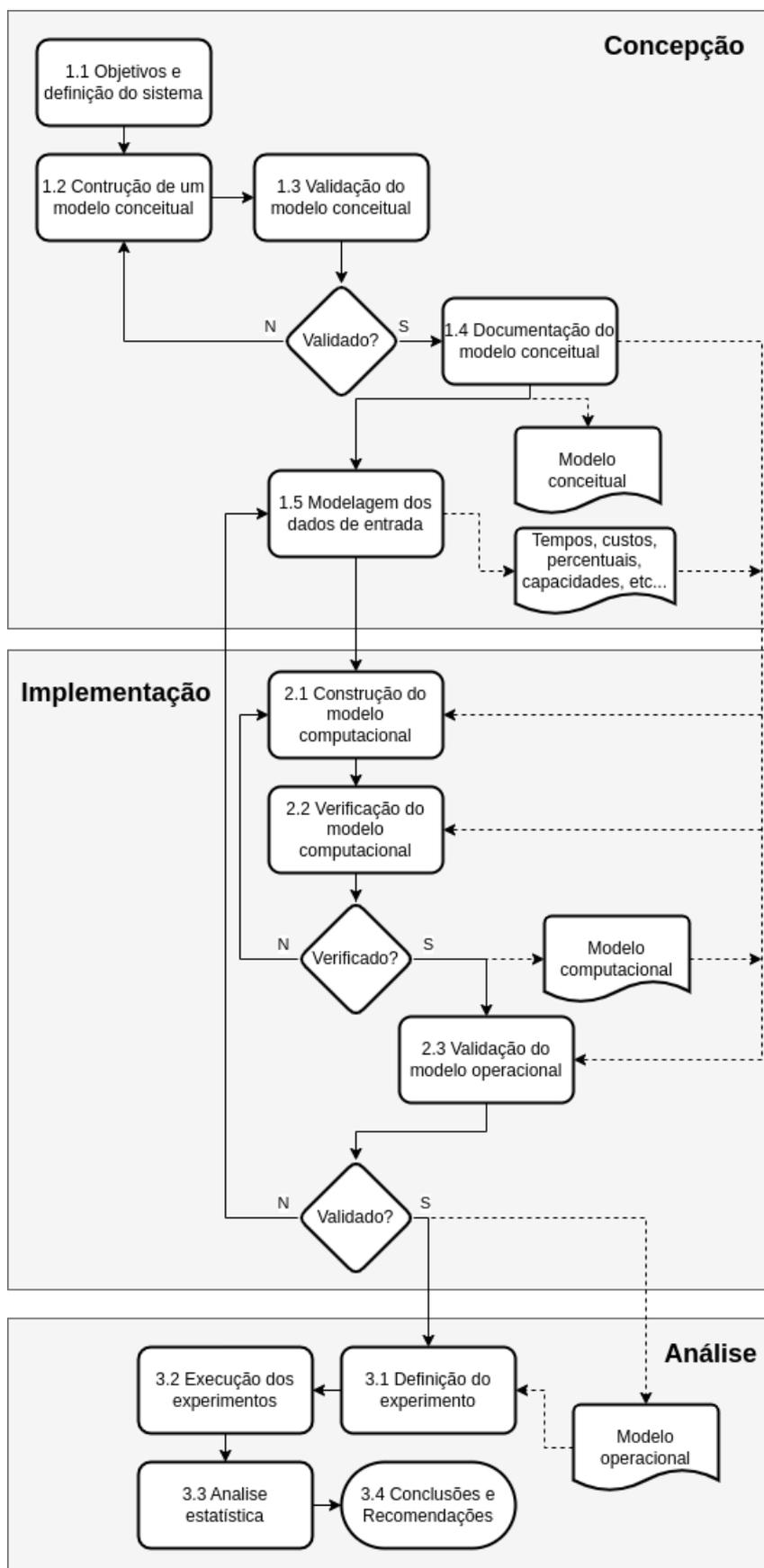
Para Montevechi *et al.* (2015) e Law e Kelton (2000), modelagem e simulação é uma poderosa ferramenta para auxílio na tomada de decisão. Com esta ferramenta é possível replicar computacionalmente sistemas complexos, permitindo que sejam feitas e estudadas intervenções sem afetar o sistema real. Um dos grandes benefícios das simulações, segundo Greasley (2004), é a redução dos riscos na tomada de decisões, uma vez que é possível quantificar as melhorias no sistema antes de sua instalação. É possível encontrar aplicações para simulações em diversos campos da indústria e da academia. Alguns exemplos notórios são: manufatura, saúde, bioengenharia e modelagem do desempenho humano (TAYLOR; ROBINSON, 2006). Há vários tipos de simulação, como simulação a eventos discretos (SED), simulação baseadas em agentes (SBA), sistemas dinâmicos, simulação de Monte Carlo, entre outros (AMARAL *et al.*, 2022; SANTOS *et al.*, 2021).

A SED se destaca na modelagem de sistemas que consistem em operações que ocorrem em momentos aleatórios e não linearmente distribuídos (O’KANE; SPENCELEY; TAYLOR, 2000; AMARAL *et al.*, 2022). Ainda segundo os autores essas duas características tornam os modelos matemáticos e demais métodos impraticáveis, o que explica a popularidade da SED para modelar sistemas de manufatura. A SED é utilizada por tomadores de decisão em diversos ramos e é uma das técnicas de pesquisa mais usadas em diversos setores (MIRANDA *et al.*, 2017).

Montevechi *et al.* (2007) dividem um projeto de simulação a eventos discretos em três etapas principais: concepção, implementação e análise. Durante a concepção é construído e validado um modelo conceitual com base nos objetivos e nos dados coletados do mundo real. Na implementação, o modelo conceitual é traduzido em um modelo computacional que após validado serve de base para um modelo operacional. Por fim, na fase de análise, os modelos produzidos nas etapas anteriores são utilizados para experimentação e produção de dados para análise de resultados. Em linhas gerais, essas três etapas estão resumidas na Figura 3.

Para Montevechi *et al.* (2015), é fundamental, em um projeto de simulação, o desenvolvimento de um modelo conceitual. Ainda segundo os autores, é a partir da documentação do

Figura 3 – Etapas de um projeto de simulação a eventos discretos



Fonte: Adaptado de Montevechi *et al.* (2010).

modelo conceitual que o modelo computacional é criado. O IDEF-SIM (*Integrated Definition Methods - Simulation*) é uma técnica para modelagem e documentação de modelos conceituais que se destaca por sua simbologia adaptada especificamente para projetos de simulação (LEAL *et al.*, 2007; MONTEVECHI *et al.*, 2010). Há quatro tipos de elementos em um modelo de SED: entidades, recursos, elementos de controle e operações (SCHRIBER; BRUNNER; SMITH, 2017). As representações no IDEF-SIM para cada um desses elementos são apresentadas no Anexo A (Simbologia do IDEF-SIM) deste trabalho.

Durante a fase conceitual de um projeto de simulação a eventos discretos, uma etapa especialmente importante para esta pesquisa é a modelagem de dados de entrada, representada pelo item 1.5 na Figura 3. Nessa etapa são coletados dados relacionados a tempos, custos, capacidades, entre outros para alimentar o modelo conceitual e podem variar de acordo com o objetivo do estudo (OLIVEIRA, 2017; BANKS, 2005). Nesta fase, a cronometragem é comumente utilizada para a determinação dos tempos (TORRES, 2017). Ainda segundo o autor, outras fontes de dados podem ser utilizadas, como sistemas de ERP (*Enterprise Resource Planning*), bases de dados, planilhas, entre outros.

De posse do modelo conceitual e dos dados de entrada modelados, um modelo computacional é construído na fase de implementação. Este modelo precisa ser verificado e em seguida validado para que se possa avançar para a etapa seguinte: a etapa de análise. Caso o modelo não seja satisfatório na verificação ou na validação, é retrabalhado até que o seja (MONTEVECHI *et al.*, 2015).

A etapa final é chamada de análise. Nesta fase é necessário definir um experimento, executá-lo, fazer as devidas análises estatísticas e a partir delas propor conclusões e recomendações para o sistema (MONTEVECHI *et al.*, 2015).

2.1.1 O fator humano em simulações

A modelagem do desempenho humano em sistemas é pesquisada há pelo menos 40 anos (BUNTING; BELYAVIN, 1999). Segundo os autores, o componente humano é importante para a modelagem de processos em todas as suas etapas. Para Baines *et al.* (2005) o desempenho dos operadores humanos têm impacto significativo na competitividade de muitos sistemas de manufatura. Segundo os autores, no entanto, os tomadores de decisão estão mal equipados para mensurar esse impacto.

As simulações são úteis para analisar o desempenho humano por meio de sua interação com o sistema (HUGHES; JIANG, 2010). No entanto, em simulações de manufaturas, podem ser obtidos resultados que não condizem com a realidade devido a uma infinidade de fatores (BAINES, 1994). Um fator relevante é que os operadores humanos nesses sistemas são modelados de maneira previsível e regular (BAINES *et al.*, 2005). Os autores destacam que frequentemente os modelos para esses operadores consideram que eles desempenham de maneira constante ao longo do dia e que iniciam e finalizam seus turnos no horário planejado. Os dados coletados nesses sistemas do mundo real, no entanto, apontam uma realidade diferente.

Baines *et al.* (2005) destacam dois benefícios principais em melhorar os modelos de desempenho humano em simulações. O primeiro é tornar possível a construção de modelos de simulação mais precisos. O segundo é incentivar os projetistas de sistemas de manufatura a criar ambientes produtivos mais amigáveis para a força de trabalho, tendo em vista que os fatores de desempenho humano são levados em consideração mais no início do desenvolvimento do projeto.

Há na literatura vários *frameworks* para representar o desempenho humano. Alguns de natureza generalista e outros que focam em aspectos bastante específicos (BAINES *et al.*, 2005; CHOI, 2018; BORGHETTI; GIAMETTA; RUSNOCK, 2017; VILELA, 2015; OLIVEIRA, 2017). Dentre eles é possível citar:

- *Field theory* (teoria dos campos): descreve o comportamento humano pela interação entre fatores relacionados ao indivíduo e seu ambiente.
- *Performance-shaping factors* (fatores que modelam desempenho): um inventário de fatores que indicam uma predisposição de trabalhadores em relação a erros e desempenho.
- *Factors predicting occupational behaviour* (fatores prevendo o comportamento ocupacional): especifica fatores básicos relacionados ao comportamento ocupacional e suas inter-relações.
- *Factors affecting quality related outcomes* (fatores afetando resultados relacionados à qualidade): modela as relações de fatores referentes ao indivíduo e à organização.
- *Integrated performance modelling environment, IPME* (ambiente integrado de modelagem de desempenho): desenvolvido para modelar a contribuição humana para o desempenho do sistema.

- *Comprehensive model of industrial work design* (modelo abrangente de projeto de trabalho industrial): modela a relação entre fatores envolvendo o ser humano, a máquina, o trabalho, o ambiente de trabalho e o esquema do trabalho.
- *Human resource management (HRM) practices and productivity* (práticas de gestão de recursos humanos e produtividade): um conjunto de técnicas de gestão de recursos humanos típicas da indústria japonesa.
- *Manufacturing system design framework* (*framework* para projeto de sistemas de produção): integra produto, processo e projeto de sistema através da ênfase em fatores humanos.
- *Physical conditions, emotional state, cognitive capabilities, social status (PECS)* (condições físicas, estado emocional, capacidades cognitivas e *status* social): *framework* baseado em agentes modelando como efeitos físicos, emocionais, cognitivos e sociais afetam o desempenho do grupo.
- *Elaborated model of work design* (modelo elaborado de projeto de trabalho): especifica cinco categorias de variáveis de projeto de trabalho que abrangem indivíduos, grupos e variáveis organizacionais.
- *Performance-shaping factors for industrial plant maintenance* (fatores que modelam desempenho para manutenção de plantas industriais): inventário de fatores que influenciam a confiabilidade humana, eficiência e carga de trabalho.
- VACP (*visual, auditory, cognitive, psychomotor*): divide os recursos do operador em sete canais distintos: visual, auditório, cognitivo, psicomotor fino, psicomotor grosseiro, comunicacional e tátil.
- NASA-TLX (índice de carga de tarefa NASA): visa quantificar os fatores que contribuem para a carga de trabalho de uma tarefa minimizando a variação entre indivíduos.
- Ritmo Circadiano: modelos que incorporam o ciclo natural de atividade, sonolência e repouso do ser humano.

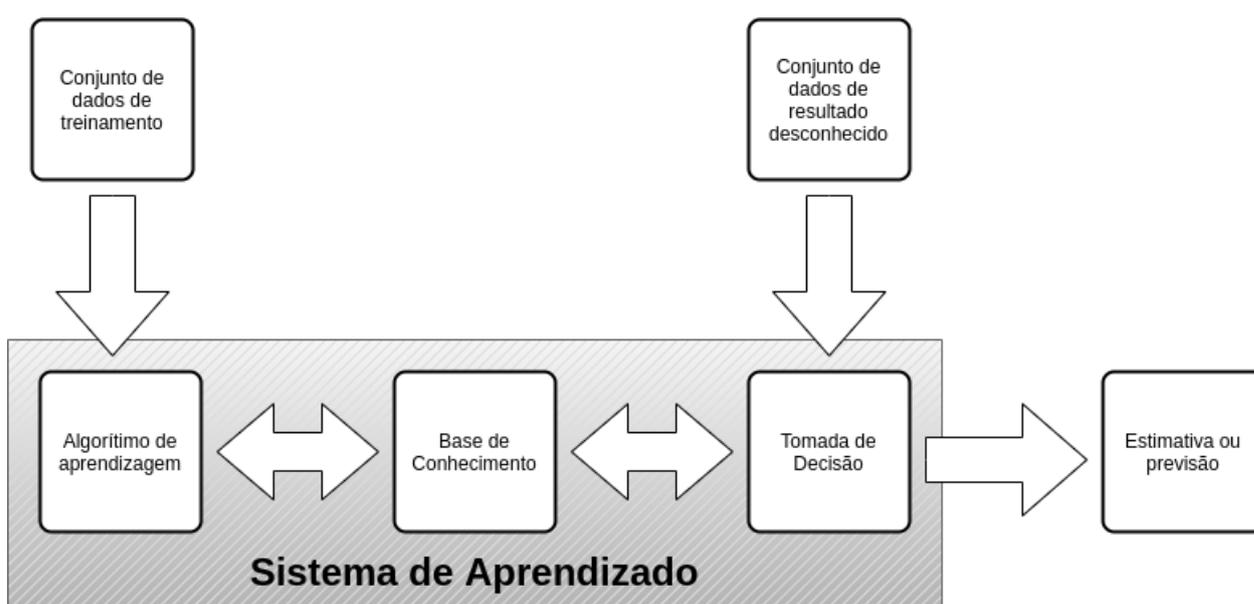
2.2 Aprendizagem de Máquina

Nilsson (1986) chama de inteligência artificial os sistemas capazes de realizar algumas das atividades mentais que os seres humanos são capazes, como por exemplo resolver problemas

matemáticos, senso comum, compreender linguagens e pilotar veículos. Nesse contexto, é importante lembrar que embora a inteligência artificial seja comumente utilizada para reproduzir atividades humanas, ela não se restringe a métodos que são observados no mundo biológico (MCCARTHY, 1998). Uma das habilidades mais fundamentais do comportamento inteligente é a habilidade de aprender (MICHALSKI; CARBONELL; MITCHELL, 1983). À reprodução dessa habilidade dentro do campo da inteligência artificial dá-se o nome Aprendizagem de Máquina (MICHALSKI; CARBONELL; MITCHELL, 1983).

Para Mohri, Rostamizadeh e Talwalkar (2012) aprendizagem de máquina se resume a produzir modelos computacionais que são capazes de aprender com a experiência, seja para melhorar seu próprio desempenho ou para produzir previsões mais precisas. Ainda segundo o autor, é importante salientar que nesse contexto, a experiência refere-se a informações do passado disponíveis para o algoritmo. Tipicamente um sistema de aprendizagem de máquina é composto por três etapas (JAIN; PAMULA; SRIVASTAVA, 2021; DU; SUN, 2006): as informações do passado são utilizadas para produzir um conjunto de dados de treinamento. Em seguida um algoritmo, ou técnica, de aprendizagem de máquina é utilizado para extrair desse conjunto uma base de conhecimento. Essa base por sua vez é utilizada em um processo de tomada de decisão que recebe um conjunto de dados cujo resultado é desconhecido e retorna uma estimativa ou previsão. A Figura 4 ilustra este processo.

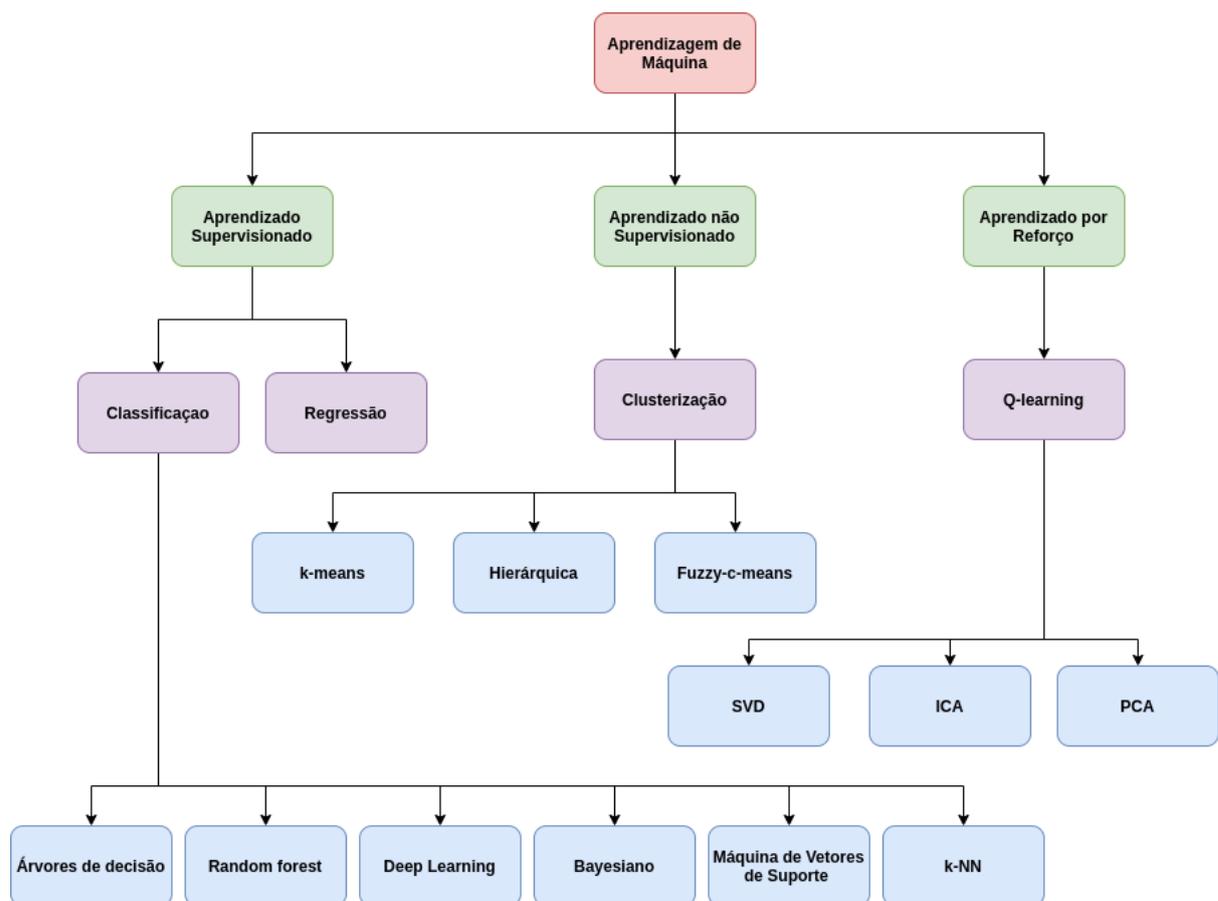
Figura 4 – Um sistema típico de aprendizagem de máquina



Fonte: Adaptado de Du e Sun (2006).

A aprendizagem de máquina pode ser segmentada de diversas formas por diferentes critérios (MICHALSKI; CARBONELL; MITCHELL, 1983, p. 7). A Figura 5 resume uma forma de se classificar as principais técnicas disponíveis. Esta pesquisa fez uso de um modelo de *Deep Learning*. Esta técnica pode ser verificada na taxonomia em aprendizado supervisionado seguido por classificação. As seções a seguir detalham cada uma dessas categorias.

Figura 5 – Taxonomia das técnicas de aprendizagem de máquina



Fonte: Adaptado de Kumar, Amgoth e Annavarapu (2019).

2.2.1 Aprendizado Supervisionado

A forma mais comum de aprendizagem de máquina é o aprendizado supervisionado (LECUN; BENGIO; HINTON, 2015). Nele o conjunto de dados de treinamento é fornecido juntamente com o resultado esperado, ou seja, a resposta para cada um dos casos (KUMAR; AMGOTH; ANNAVARAPU, 2019; MOHRI; ROSTAMIZADEH; TALWALKAR, 2012). Ainda segundo os autores, a relação entre os dados de entrada e seus respectivos resultados esperados é estimada durante o processo de treinamento, produzindo uma função que recebe um dado de

entrada e retorna a melhor estimativa para o resultado. A maior responsabilidade desse tipo de aprendizado é gerar um modelo que represente as relações e dependências entre os dados de entrada e suas respectivas previsões (KUMAR; AMGOTH; ANNAVARAPU, 2019).

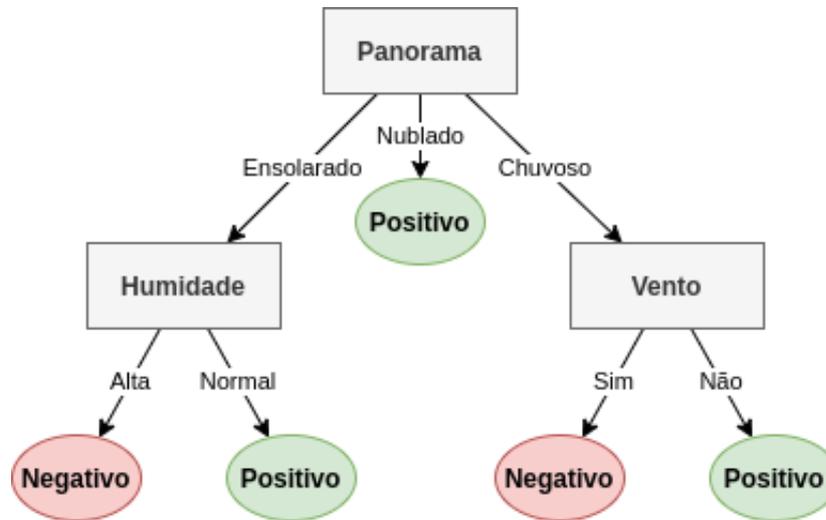
O aprendizado supervisionado é comumente utilizado para resolver problemas de classificação e regressão (KUMAR; AMGOTH; ANNAVARAPU, 2019; MOHRI; ROSTAMIZADEH; TALWALKAR, 2012). Mohri, Rostamizadeh e Talwalkar (2012) também citam a aplicação para problemas de ranqueamento. Kumar, Amgoth e Annavarapu (2019) mencionam sete diferentes técnicas de classificação: Árvores de Decisão, *Random Forest* (do inglês, floresta aleatória), Redes Neurais Artificiais, *Deep Learning* (do inglês, aprendizado profundo), Aprendizado Bayesiano, Máquina de Vetores de Suporte e Vizinheiro Mais Próximo (k-NN).

Segundo Montgomery (2012), regressão é uma técnica estatística para investigar e modelar a relação entre variáveis. Regressão é um tipo de aprendizado supervisionado em que tenta prever um valor (Y) em função de um conjunto de características (X) (KUMAR; AMGOTH; ANNAVARAPU, 2019). As variáveis nesse contexto são contínuas ou quantitativas. A Equação 2.1 apresenta a notação matemática para regressões (KUMAR; AMGOTH; ANNAVARAPU, 2019). Nela observa-se que Y, a variável de saída, é dependente de X, a variável de entrada. Enquanto ε representa o erro aleatório do modelo de regressão (KUMAR; AMGOTH; ANNAVARAPU, 2019).

$$Y = f(X) + \varepsilon \quad (2.1)$$

Árvores de decisão compõem uma classe de estratégias de aprendizagem de máquina em que a base de conhecimento é representada por uma árvore de condições do tipo "se X então Y" (QUINLAN, 1986). Essas condições são em geral representadas em árvores para melhor legibilidade (DU; SUN, 2006). Cada condição divide o conjunto de dados a ser classificado em subconjuntos menores até que um conjunto final, representando a resposta, seja alcançado (DU; SUN, 2006). Cada um desses conjuntos é chamado de nó dentro da árvore (QUINLAN, 1986). Os nós intermediários são denominados nós de decisão (QUINLAN, 1986). Já os nós finais são denominados nós folha, seguindo a analogia com árvores (QUINLAN, 1986). A Figura 6 apresenta um exemplo simplificado dessa representação. Os nós retangulares são os nós de decisão. Já os nós redondos são nós folha.

Figura 6 – Árvore de decisão simples



Fonte: Adaptado de Quinlan (1986).

Random Forest, do inglês Floresta Aleatória, é uma técnica de aprendizagem de máquina supervisionada que consiste em uma combinação de árvores de decisão em que cada decisão depende de valores aleatórios. O algoritmo seleciona e gera novas árvores com o objetivo de minimizar o erro dentro dos dados de treinamento. Cada árvore então vota no resultado mais popular (BREIMAN, 2001). A técnica *Random Forest* é eficiente quando aplicada a conjuntos de dados volumosos e esparsos (KUMAR; AMGOTH; ANNAVARAPU, 2019).

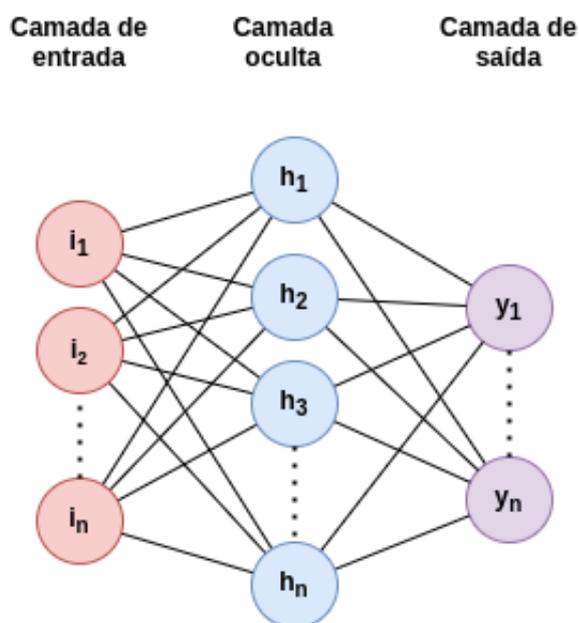
Uma rede neural artificial é uma técnica de aprendizagem de máquina inspirada nos neurônios biológicos que consiste em elementos de processamento, chamados de neurônios, e conexões entre eles (SHANMUGANATHAN, 2016). A Equação 2.2 representa o funcionamento típico de um neurônio onde y é o valor de saída do neurônio, i_n representa os sinais de entrada, w_n seus respectivos pesos e t o limiar de ativação do neurônio.

$$y = \begin{cases} 0 & \text{se } \sum_{n=1}^m i_n \times w_n < t \\ 1 & \text{se } \sum_{n=1}^m i_n \times w_n \geq t \end{cases} \quad (2.2)$$

A Equação 2.2 mostra que o sinal de saída do neurônio é calculado a partir da soma dos sinais de entrada multiplicados a seus respectivos pesos. Caso o valor calculado ultrapasse um limiar determinado, o neurônio emite um sinal positivo a diante. Do contrário, um sinal de valor zero é emitido (SHANMUGANATHAN, 2016).

Redes neurais são construídas conectando-se uma camada de neurônios de entrada a uma camada de saída. Opcionalmente uma ou mais camadas intermediárias, ou camadas ocultas, podem ser inseridas entre as camadas de entrada e de saída (GARDNER; DORLING, 1998). A Figura 7 mostra esse arranjo de forma simplificada.

Figura 7 – Uma rede neural simples



Fonte: Adaptado de Kumar, Amgoth e Annavarapu (2019).

Deep Learning, do inglês Aprendizado Profundo, é uma subcategoria das redes neurais artificiais. Segundo LeCun, Bengio e Hinton (2015) *Deep Learning* é capaz de superar uma limitação das demais técnicas de aprendizagem de máquina, incluindo redes neurais artificiais: *Deep Learning* é capaz de processar dados em sua forma bruta. Ainda segundo o autor, técnicas tradicionais de aprendizagem de máquina necessitam considerável domínio e expertise para construção de um extrator de características (em inglês *feature extractor*) que transforma os dados brutos em dados que o algoritmo de aprendizagem é capaz de processar. Em *Deep Learning* são utilizados múltiplos níveis de módulos não lineares que executam tanto a extração de características dos dados brutos quanto a classificação, estimativa ou previsão desejada (LECUN; BENGIO; HINTON, 2015).

O Aprendizado Bayesiano é uma técnica de aprendizado supervisionado que se diferencia das técnicas tradicionais por utilizar probabilidades para expressar todas as formas de incerteza (NEAL, 1996). Ainda segundo o autor, o aprendizado nesse contexto ocorre por aplicação de

regras de probabilidade. O Aprendizado Bayesiano encontra relações entre os conjuntos de dados aplicando independência condicional usando vários métodos estatísticos (KUMAR; AMGOTH; ANNAVARAPU, 2019).

Vizinho Mais Próximo, ou k-NN, do inglês *k-Nearest Neighbor*, é uma das técnicas de aprendizagem de máquina mais diretas e simples (KUMAR; AMGOTH; ANNAVARAPU, 2019). Ainda segundo os autores, nessa técnica é utilizada uma função de distância entre as entradas do espaço amostral, tipicamente uma distância euclidiana. Calcula-se a distância de cada entrada para a que se deseja estimar, selecionando um número k de entradas mais próximas. A classificação conhecida de cada uma das amostras mais próximas conta como um voto para o resultado. O resultado mais votado é o resultado da estimativa.

2.2.2 Aprendizado não supervisionado

No aprendizado não supervisionado, o aprendiz processa os dados sem conhecer seu resultado esperado e faz previsões sobre os pontos não conhecidos. Uma vez que não se conhece os resultados corretos para nenhum dos dados de entrada, nesse tipo de aprendizado, é difícil quantificar ou validar o desempenho do aprendiz (MOHRI; ROSTAMIZADEH; TALWALKAR, 2012).

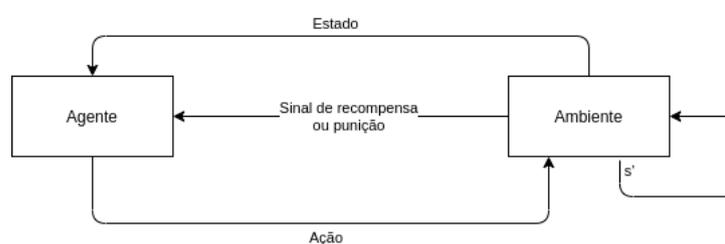
2.2.3 Aprendizado por reforço

No aprendizado por reforço as fases de treinamento e teste se misturam (MOHRI; ROSTAMIZADEH; TALWALKAR, 2012). Para os autores, o aprendizado se dá através da interação com o ambiente e subsequente coleta de *feedback* quanto ao efeito dessa interação. Os seres humanos estão constantemente observando a correlação entre suas ações e os resultados produzidos para obter conhecimento. No mundo artificial, é quantificada uma recompensa numérica que incentiva ou desestimula determinado comportamento no agente. Essa técnica é chamada de Aprendizado por Reforço, do inglês *Reinforcement Learning* (SUTTON; BARTO, 2018; MOHRI; ROSTAMIZADEH; TALWALKAR, 2012).

A aprendizagem por reforço acontece quando uma agente precisa aprender através de tentativa e erro em um ambiente dinâmico (KAELBLING; LITTMAN; MOORE, 1996). O modelo de Aprendizado por Reforço representado na Figura 8 se resume à repetição dos quatro passos abaixo (QIANG; ZHONGLI, 2011):

1. O agente primeiro observa um estado.
2. A partir deste estado realiza uma ação.
3. Essa ação produz uma mudança no estado do ambiente (s').
4. O agente observa essa atualização no estado do ambiente e calcula o valor da recompensa ou punição sobre a ação que tomou.

Figura 8 – Modelo básico de Aprendizado por Reforço



Fonte: Adaptado de (QIANG; ZHONGLI, 2011).

O objetivo de um algoritmo de Aprendizagem por Reforço é encontrar uma estratégia que produza a maior recompensa cumulativa possível em (QIANG; ZHONGLI, 2011). Alguns dos principais algoritmos desenvolvidos com esse propósito são: o algoritmo de Sarsa, o algoritmo da diferença temporal, o algoritmo de aprendizagem Q e o algoritmo de aproximação de função (QIANG; ZHONGLI, 2011).

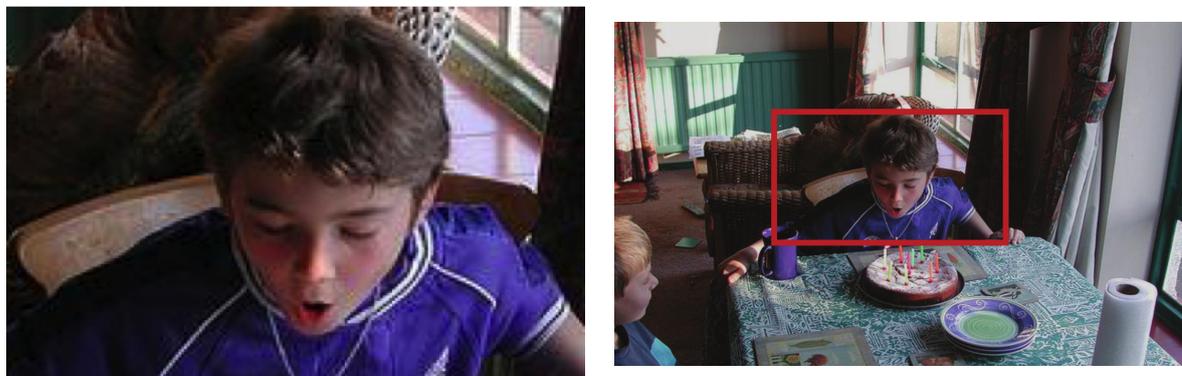
2.3 Detecção de Humor

2.3.1 O modelo EMOTIC

Kosti *et al.* (2019) apresentam o modelo EMOTIC (*emotion in context* do inglês "emoção em contexto"), que é um modelo de aprendizagem de *Deep Learning* para classificar o humor aparente de indivíduos em uma imagem. O modelo é treinado a partir do *Emotions in Context (EMOTIC) Dataset*, um conjunto de imagens de pessoas em ambientes reais com anotações referentes a suas emoções aparentes (KOSTI *et al.*, 2017). Este *dataset* possui o diferencial de levar em consideração além das expressões do indivíduo o contexto em que ele se encontra. Segundo os autores, o contexto é fundamental para que se identifique corretamente as emoções

em uma imagem. Tal constatação fica evidente quando comparadas as emoções aparentes na Figura 9a e na Figura 9b.

Figura 9 – Comparação de imagens com e sem contexto



(a) Imagem com pouco contexto

(b) Imagem em contexto

Fonte: (KOSTI *et al.*, 2019).

Foram definidas para o *EMOTIC Dataset* 26 categorias discretas para a emoção aparente e 3 contínuas. Cada indivíduo em uma imagem pode ser classificado em uma ou mais categorias discretas. Já para as categorias contínuas, os indivíduos são classificados com um valor de 1 a 10. A Tabela 1 apresenta estas categorias (KOSTI *et al.*, 2017).

Tabela 1 – Variáveis do EMOTIC.

Variável	Tipo	Tradução	Descrição
<i>Valence</i>	Contínua	Valência	mensura o quão positiva ou prazerosa uma emoção é
<i>Arousal</i>	Contínua	Excitação	mensura o grau de agitação da pessoa
<i>Dominance</i>	Contínua	Domínio	mensura o quão em controle da situação a pessoa está
<i>Peace</i>	Discreta	Paz	está bem e relaxado; sem preocupações; têm pensamentos ou sentimentos positivos; satisfeito
<i>Affection</i>	Discreta	Afeição	sentimento afetuoso; amor; ternura
<i>Esteem</i>	Discreta	Estima	sentimentos de opinião ou julgamento favorável; respeito; admiração; apreciação; gratidão
<i>Anticipation</i>	Discreta	Antecipação	estado que procura avanço; esperando ou se preparando para possíveis eventos futuros

<i>Engagement</i>	Discreta	Engajamento	prestando atenção a algo; absorto em algo; curioso; interessado
<i>Confidence</i>	Discreta	Confiança	sentimento de estar certo; convicção de que o resultado será favorável; encorajado; orgulhoso
<i>Happiness</i>	Discreta	Felicidade	sentindo encantado; sentindo prazer ou diversão
<i>Pleasure</i>	Discreta	Prazer	sensação de deleite
<i>Excitement</i>	Discreta	Entusiasmo	sentir entusiasmo; estimulado; energético
<i>Surprise</i>	Discreta	Surpresa	a descoberta repentina de algo inesperado
<i>Sympathy</i>	Discreta	Empatia	estado de compartilhar outras emoções, objetivos ou problemas; suporte; compassivo
<i>Doubt/Confusion</i>	Discreta	Dúvida/Confusão	dificuldade de compreender ou decidir; pensando em opções diferentes
<i>Disconnection</i>	Discreta	Desconexão	sentindo desinteressado dos eventos ao redor; indiferente; indiferente; entediado; distraído
<i>Fatigue</i>	Discreta	Fadiga	desgaste; cansaço; com sono
<i>Embarrassment</i>	Discreta	Constrangimento	sentindo envergonhado ou culpado
<i>Yearning</i>	Discreta	Anseio	forte desejo de ter algo; ciúme; inveja; luxúria
<i>Disapproval</i>	Discreta	Desaprovação	sentimento de que algo está errado ou repreensível; descontente; hostil
<i>Aversion</i>	Discreta	Aversão	sentir nojo; antipatia; repulsa; sentindo ódio
<i>Annoyance</i>	Discreta	Incômodo	incomodado por algo ou alguém; irritado; impaciente; frustrado
<i>Anger</i>	Discreta	Ira	intenso desprazer ou raiva; furioso; ressentido
<i>Sensitivity</i>	Discreta	Sensibilidade	sensação de estar fisicamente ou emocionalmente ferido; sentindo delicado ou vulnerável
<i>Sadness</i>	Discreta	Tristeza	sentindo infeliz; triste; desapontado ou desanimado
<i>Disquietment</i>	Discreta	Inquietação	nervoso; preocupado; chateado; ansioso; tenso; pressionado; alarmado

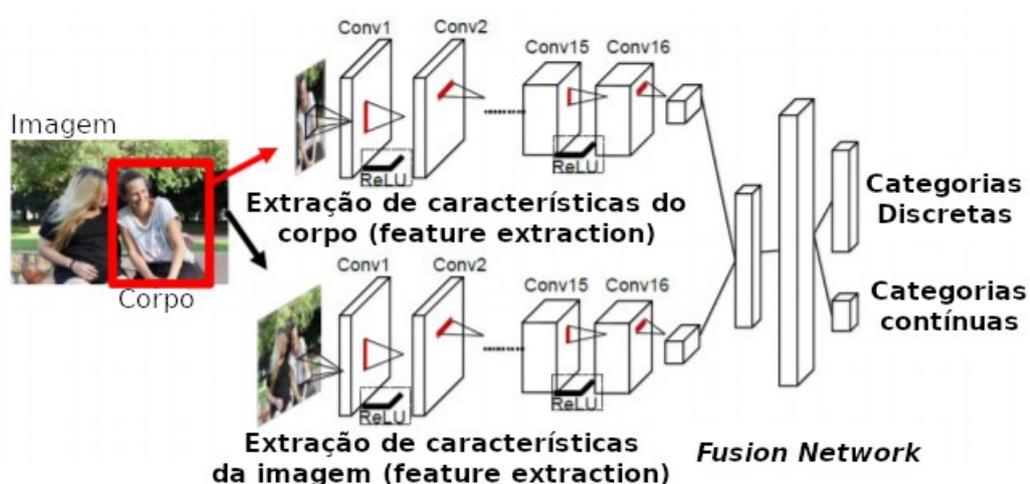
<i>Fear</i>	Discreta	Medo	sentindo desconfiado ou com medo do perigo; ameaçado; mal ou dor; horrorizado
<i>Pain</i>	Discreta	Dor	sofrimento físico
<i>Suffering</i>	Discreta	Sufrimento	dor psicológica ou emocional; agoniado/perturbado; angustiado

Fonte: Adaptado de Kosti *et al.* (2017)

Cada uma destas imagens foi então anotada manualmente para os critérios acima. O *EMOTIC Dataset* consiste neste conjunto de imagens associadas a suas respectivas anotações.

Utilizando o *EMOTIC Dataset* como base para treinamento, foi criado o modelo EMOTIC. Como o diferencial do modelo EMOTIC para os demais é o fato de levar em consideração o contexto da imagem além da expressão do indivíduo, é extraído o retângulo que contém a pessoa da imagem. O retângulo extraído alimenta uma série de camadas de redes neurais convolucionais para extração de atributos. A imagem completa alimenta uma segunda série de camadas de redes neurais convolucionais para extração de atributos do contexto. Ambos os resultados são conectados a uma última série de camadas que calculam os resultados. A Figura 10 resume este arranjo.

Figura 10 – Modelo EMOTIC



Fonte: Adaptado de Kosti *et al.* (2019)

Os autores disponibilizam o modelo EMOTIC pré treinado e pronto para uso. Este modelo inclui o código fonte e parâmetros ajustados para o modelo de *deep learning*.

3 REVISÃO SISTEMÁTICA DA LITERATURA

Neste capítulo é apresentada uma revisão sistemática de literatura (RSL) sobre como o desempenho humano é representado em modelos de simulação.

3.1 Método de Revisão Sistemática da Literatura

A revisão da literatura é uma ferramenta de grande importância para o avanço do conhecimento (LEVY; ELLIS, 2006; XIAO; WATSON, 2017). Ainda segundo os autores, uma revisão da literatura permite que pesquisadores identifiquem onde está o limite atual do conhecimento antes de expandi-lo. Segundo Oliveira, Lima e Montevechi (2016) a RSL tem como vantagem sobre perante uma revisão da literatura de abordagem não sistemática uma redução na probabilidade de haver tendências e distorções nos resultados da pesquisa. Isso se deve à fase de pesquisa ser feita de forma mais objetiva, minimizando a interferência do pesquisador.

Denyer e Tranfield (2008 apud AMARAL; LIMA; MIRANDA, 2020) propõem os seguintes procedimentos para elaboração de uma RSL:

1. **Projeto:** nessa etapa, as perguntas de pesquisa e os critérios de limitação da pesquisa são definidos, incluindo as bases e critérios de pesquisa que serão utilizados.
2. **Busca e Seleção:** após a definição do projeto, é realizada a busca nas bases de dados selecionadas, seguindo os critérios definidos anteriormente.
3. **Análise e Síntese:** nesta etapa, é feita a triagem dos estudos identificados com base em sua relevância para a questão da pesquisa e os critérios de inclusão. Os títulos e resumos são sumarizados e classificados quanto a padrões e tendências identificadas na literatura.
4. **Apresentação dos Resultados:** por fim, são apresentadas as tendências e padrões encontrados na análise e síntese dos estudos, e as conclusões do estudo são traçadas.

Os procedimentos realizados em cada uma dessas etapas estão detalhados nas seções a seguir.

3.1.1 Projeto

Essa RSL utilizou como base de periódicos o Scopus, fornecido pela Elsevier e disponibilizado para instituições a partir do portal de periódicos da Capes. O Scopus é uma base contendo mais de 22 mil títulos e mais de 5 mil editores ao redor do mundo (ELSEVIER, 2015). A base foi escolhida pela disponibilidade para os autores e pela excelente ferramenta de pesquisa.

Na intenção de explorar como o desempenho humano é representado em simulações dentro da literatura, as seguintes perguntas de pesquisa foram estabelecidas:

1. Quais fatores para o desempenho humano são mais comumente representados em simulações?
2. Quais técnicas são utilizadas para modelar esses fatores?
3. Que lacunas na literatura podem ser identificadas através dessa revisão?

3.1.2 Busca e Seleção

Os artigos foram selecionados para a RSL a partir dos seguintes critérios:

- O documento é um artigo publicado em revista científica.
- O artigo está disponível no Scopus.
- O artigo foi escrito em inglês.
- O tema do artigo é relacionado a "Simulação" e "Desempenho Humano".

Para fins de pesquisa, um artigo é considerado de um tema caso contenha um termo relacionado em seu título, palavras chaves ou resumo. Os termos relacionados a simulação utilizados foram: "*Discrete Event Simulation*" (do inglês, Simulação a Eventos Discretos), "*Agent Based Simulation*" (do inglês, Simulação Baseada em Agentes) e "*Hybrid Simulation*" (do inglês, Simulação Híbrida). Para desempenho humano, apenas o termo em inglês, "*Human Performance*" foi utilizado. A combinação destes termos e critérios produziu a seguinte chave de pesquisa:

```
(TITLE-ABS-KEY("discrete event simulation") OR  
TITLE-ABS-KEY("agent based simulation") OR
```

```
TITLE-ABS-KEY("hybrid simulation"))
AND (TITLE-ABS-KEY("Human performance"))
AND LANGUAGE(english) AND DOCTYPE(ar)
```

A pesquisa na base de periódicos retornou um total de 18 artigos que satisfazem a todos os critérios definidos. Uma análise qualitativa manual foi realizada através da leitura do título e resumo de cada um dos artigos. Após essa análise, 9 artigos foram classificados como referentes ao tema pesquisados e os outros 9 descartados por não estarem relacionados ao tema da revisão.

3.1.3 Análise e Síntese

Os dados referentes aos artigos retornados pela busca foram baixados do Scopus e importados em uma planilha eletrônica. Nessa planilha foram demarcados e filtrados os artigos que foram rejeitados pela análise qualitativa. Em seguida os atributos qualitativos extraídos da leitura de cada um dos artigos foram adicionados à planilha para serem analisados.

3.1.4 Apresentação dos Resultados

Os resultados foram tabulados de acordo com cada critério e apresentados em ordem de relevância. Nessas tabelas foi possível observar padrões nos artigos revisados, como que em tipos de simulação o desempenho humano é mais frequentemente modelado bem como quais fatores humanos e quais técnicas são utilizadas na modelagem. Também foi possível inferir algumas possíveis lacunas na literatura sobre o assunto.

3.2 Resultados e discussões

Os principais resultados extraídos desta pesquisa encontram-se resumidos na Tabela 2 abaixo:

Tabela 2 – Resultados da RSL

Perspectiva	Nº Artigos	%	Citações
Tipo de Simulação			
SED	6	66,67%	Majdzik (2022), Vilela <i>et al.</i> (2020), Borghetti, Giametta e Rusnock (2017), Paiva <i>et al.</i> (2017), Mason <i>et al.</i> (2005) e Baines e Kay (2002)

SBA	2	22,22%	Choi (2018) e Gregoriades <i>et al.</i> (2010)
Simulação Híbrida (SED e Sistemas Dinamicos)	1	11,11%	Zhang <i>et al.</i> (2017)
Aplicação			
Linha de montagem	5	55,56%	Majdzik (2022), Vilela <i>et al.</i> (2020), Paiva <i>et al.</i> (2017), Mason <i>et al.</i> (2005) e Baines e Kay (2002)
Supervisão	1	11,11%	Choi (2018)
Alocação de tarefas	1	11,11%	Zhang <i>et al.</i> (2017)
Projeto de rodovias	1	11,11%	Gregoriades <i>et al.</i> (2010)
Fator que Influencia o Desempenho Humano Considerado			
Tempo em Jornada	4	44,44%	Majdzik (2022), Vilela <i>et al.</i> (2020), Paiva <i>et al.</i> (2017) e Mason <i>et al.</i> (2005) Choi (2018), Zhang <i>et al.</i> (2017),
Carga de trabalho	4	44,44%	Borghetti, Giametta e Rusnock (2017) e Gregoriades <i>et al.</i> (2010)
Qualificação ou Experiência	1	11,11%	Majdzik (2022)
Atenção	1	11,11%	Gregoriades <i>et al.</i> (2010)
Variação Individual	1	11,11%	Mason <i>et al.</i> (2005)
Traços de personalidade	1	11,11%	Baines e Kay (2002)
Ambiente	1	11,11%	Baines e Kay (2002)
<i>Operator Functional State</i> (OFS)	1	11,11%	Zhang <i>et al.</i> (2017)
Técnica para modelar o desempenho humano			
Modelo Fuzzy	2	22,22%	Majdzik (2022) e Zhang <i>et al.</i> (2017)
<i>Fit</i> Estatístico	2	22,22%	Vilela <i>et al.</i> (2020) e Mason <i>et al.</i> (2005)
VACP	2	22,22%	Choi (2018) e Borghetti, Giametta e Rusnock (2017)
Eletroencefalograma	2	22,22%	Zhang <i>et al.</i> (2017) e Borghetti, Giametta e Rusnock (2017)
Ritmo circadiano	1	11,11%	Paiva <i>et al.</i> (2017)
Redes Bayesianas	1	11,11%	Gregoriades <i>et al.</i> (2010)
Coefficiente de variação	1	11,11%	Mason <i>et al.</i> (2005)
Rede Neural Artificial	1	11,11%	Baines e Kay (2002)
HIP	1	11,11%	Choi (2018)
IMPRINT	1	11,11%	Borghetti, Giametta e Rusnock (2017)
NASA-TLX	1	11,11%	Borghetti, Giametta e Rusnock (2017)

Fonte: Autores

As seções a seguir apresentam uma visão geral das características dos trabalhos revisados nesta pesquisa.

3.2.1 Fatores Humanos Considerados

Os principais fatores que impactam o desempenho humano considerados na literatura analisada são o tempo em jornada e a carga de trabalho. Embora haja uma grande variedade na forma como esses fatores são definidos e modelados, de forma geral, o tempo em jornada refere-se ao tempo decorrido desde que o operador iniciou a jornada de trabalho. Esse tempo, na maioria das representações é representado em categorias discretas (MAJDZIK, 2022; VILELA *et al.*, 2020; PAIVA *et al.*, 2017), porém há casos em que o fator é representado de forma contínua (MASON *et al.*, 2005).

A carga de trabalho, por sua vez, representa o quanto dos recursos físicos, motores e cognitivos do operador estão sendo utilizados para realizar a tarefa (CHOI, 2018; BORGHETTI; GIAMETTA; RUSNOCK, 2017). Esse fator está presente em todos os artigos envolvendo Simulação Baseada em Agentes (SBA) revisados e pode também ser encontrado em artigos sobre Simulação a Eventos Discretos (SED).

3.2.2 Técnicas para modelar o desempenho humano

Foi encontrada na literatura uma grande variedade de técnicas para modelar o desempenho humano em simulações. Vilela *et al.* (2020) e Paiva *et al.* (2017) representam a variação do desempenho dentro da jornada de trabalho atribuindo diferentes curvas de distribuição de probabilidade a faixas de tempo dentro da jornada. Esta diferença permitiu diminuir o erro nos resultados simulados em relação ao sistema real. Mason *et al.* (2005), por sua vez, fazem um fit estatístico da curva de variação de desempenho do operador ao longo da jornada de trabalho e aplicam um coeficiente de variação para representar a diferença de desempenho entre indivíduos.

Uma outra ferramenta promissora para representar o desempenho humano em simulações são os modelos *fuzzy*. Esses modelos permitem representar e relacionar matematicamente mais de um fator de desempenho ao mesmo tempo, como a variação devido ao tempo de jornada, a carga de trabalho e a experiência relativa dos operadores. (MAJDZIK, 2022; ZHANG *et al.*, 2017).

Gregoriades *et al.* (2010) utilizam redes bayesianas de crença (*bayesian belief networks*)

para inferir dois fatores chave: carga de trabalho e atenção do operador. Esses fatores chave são inferidos a partir de vários outros fatores calculados em uma SBA.

A neurociência também pode ser uma aliada na representação desses fatores. Eletroencefalogramas em conjunto com outras tecnologias, como redes neurais artificiais e modelos *fuzzy* podem ser utilizados, por exemplo, para extrair dos operadores a carga de trabalho ao executar determinada tarefa (ZHANG *et al.*, 2017; BORGHETTI; GIAMETTA; RUSNOCK, 2017).

Existe também uma variedade de modelos para representar a carga de trabalho dos operadores na literatura. Dentre eles destaca-se a escala VACP, *visual, auditory, cognitive, psychomotor*, que divide os recursos do operador em sete canais distintos: visual, auditório, cognitivo, psicomotor fino, psicomotor grosseiro, comunicacional e tátil (BORGHETTI; GIAMETTA; RUSNOCK, 2017). Uma outra técnica que visa quantificar a carga de trabalho de tarefas é o índice de carga de tarefa NASA (NASA-TLX). Segundo Hart e Staveland (1988), o NASA-TLX visa quantificar os fatores que contribuem para a carga de trabalho de uma tarefa minimizando a variação entre indivíduos.

3.2.3 Lacunas na Literatura

O volume pequeno de trabalhos retornados pela pesquisa é um indício de que, no geral, a literatura para o tema é escassa e carece de uma exploração mais ampla. Além disso, algumas lacunas específicas foram identificadas dentro do padrão de artigos estudado.

A grande maioria dos artigos representando o desempenho humano em simulações utilizam SED. As SBAs, embora presentes, estão sub-representadas e poderiam ser melhor exploradas. Essa RSL também evidencia que os fatores humanos modelados em SED são diferentes dos modelados em SBA. Uma possível lacuna portanto está em explorar e comparar esses modelos em SED e SBA.

Para Digiesi *et al.* (2009), o comportamento humano e seu desempenho podem ser afetados por diversos fatores: o ambiente de trabalho (físico: microclima, ergonomia e barulho; social: relações humanas e comunicação com o grupo), a natureza da tarefa (discreta ou contínua e motora ou cognitiva) e fatores pessoais (atitude psicofísica, habilidade pessoal, idade e sexo). No entanto, nenhum artigo abordando o ambiente social de trabalho foi encontrado nesta revisão.

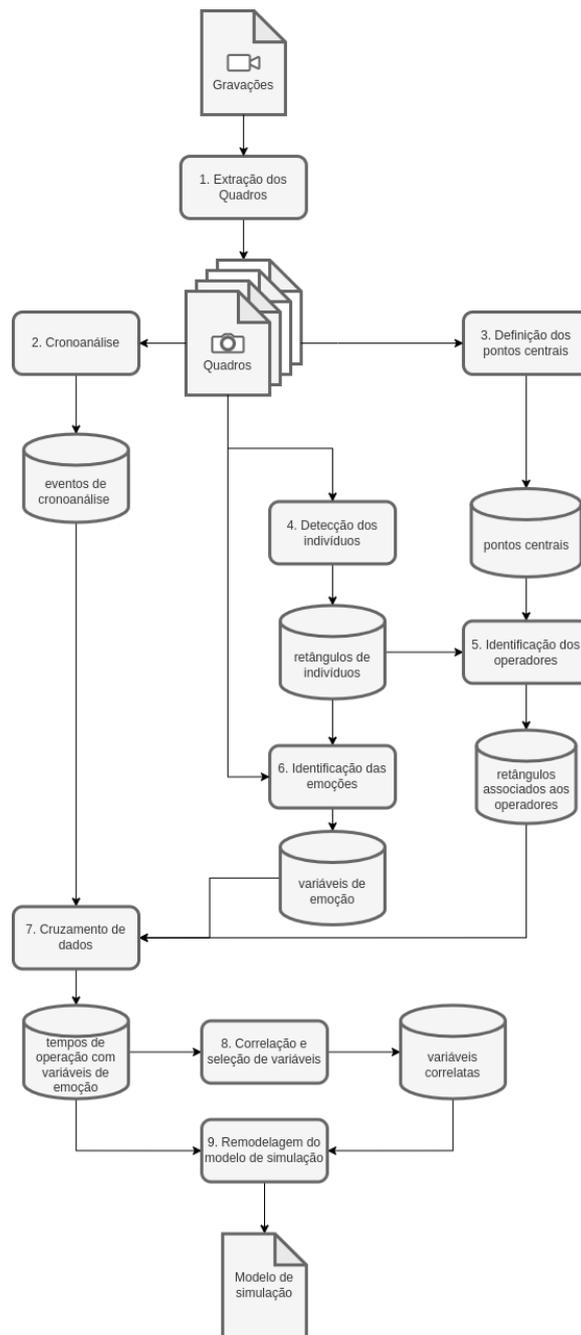
Em relação às técnicas e tecnologias utilizadas, para alguns trabalhos foi necessário a utilização de eletroencefalogramas para que fosse possível modelar alguns fatores (ZHANG *et al.*,

2017; BORGHETTI; GIAMETTA; RUSNOCK, 2017). Esta pesquisa apresenta uma alternativa mais acessível para modelar um fator pessoal. Utilizando um modelo de *Deep Learning*, esta pesquisa modela o fator humor em um modelo de simulação necessitando apenas das filmagens da linha de montagem em operação.

4 DESENVOLVIMENTO E DISCUSSÕES

Neste capítulo, as etapas descritas na subseção 1.2.2 são descritas conforme foram executadas. A Figura 11 resume como os dados foram transformados e relacionados para obter os resultados desta pesquisa.

Figura 11 – Diagrama de fluxo de dados



Fonte: Autor

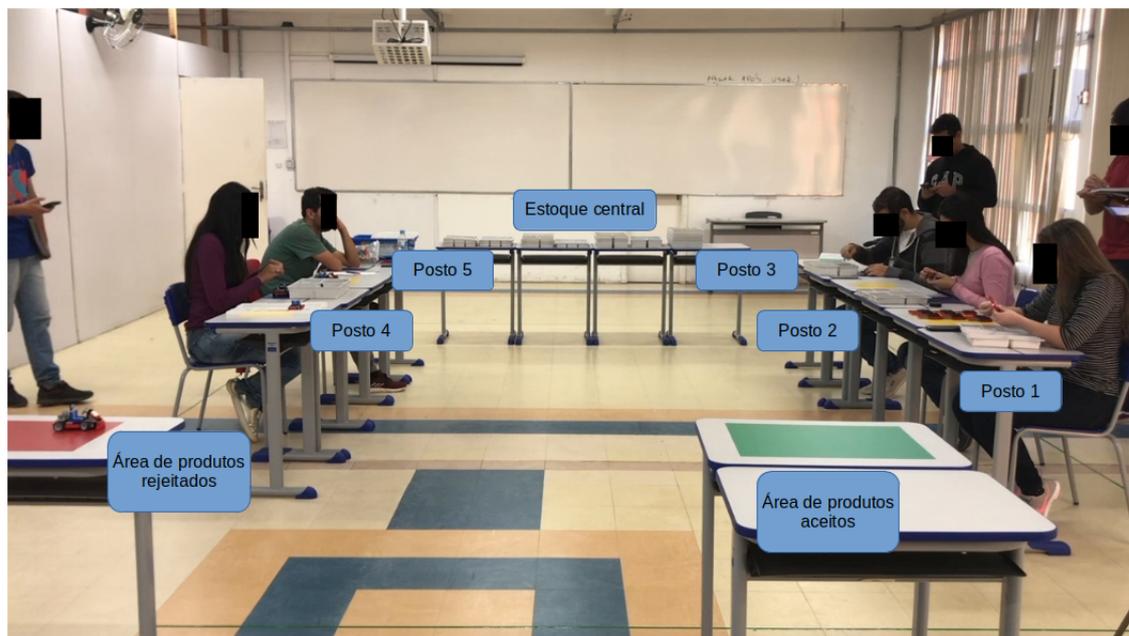
Os passos apresentados na Figura 2 são apresentados sequencialmente nas seções a seguir. Cada seção deste capítulo pode ser associada diretamente a um ou mais itens do diagrama. Cabe observar que os processos iterativos foram condensados em uma única seção para cada.

4.1 Preparação dos dados

4.1.1 Obtenção das gravações

A amostra desta pesquisa consistiu nas gravações da linha de montagem de *Karts Lego*[®] em operação da pesquisa de Rocha (2020). Trata-se de uma linha de produção fictícia criada para o ensino de cronoanálise. Durante a pesquisa da autora, esta linha foi colocada em funcionamento e operada por pesquisadores e estudantes da Universidade Federal de Itajubá. A operação foi filmada por duas câmeras durante 14 ciclos com peças suficientes para montar 40 *Karts* em cada ciclo. Ao final foram produzidas 28 horas de gravação bem como dados cronometrados de tempos de operação e reposição para cada posto de trabalho da linha.

A linha de montagem de *Karts Lego*[®] é composta por cinco postos de trabalho, sendo os quatro primeiros para as etapas de montagem e o quinto para inspeção. A linha de montagem possui um estoque central de matéria prima. Cada posto de trabalho possui seu estoque de matéria prima que pode ser repostado pelo próprio operador do estoque central quando necessário em lotes de oito unidades. Entre cada posto de trabalho há um estoque intermediário de material semi acabado (ROCHA, 2020).

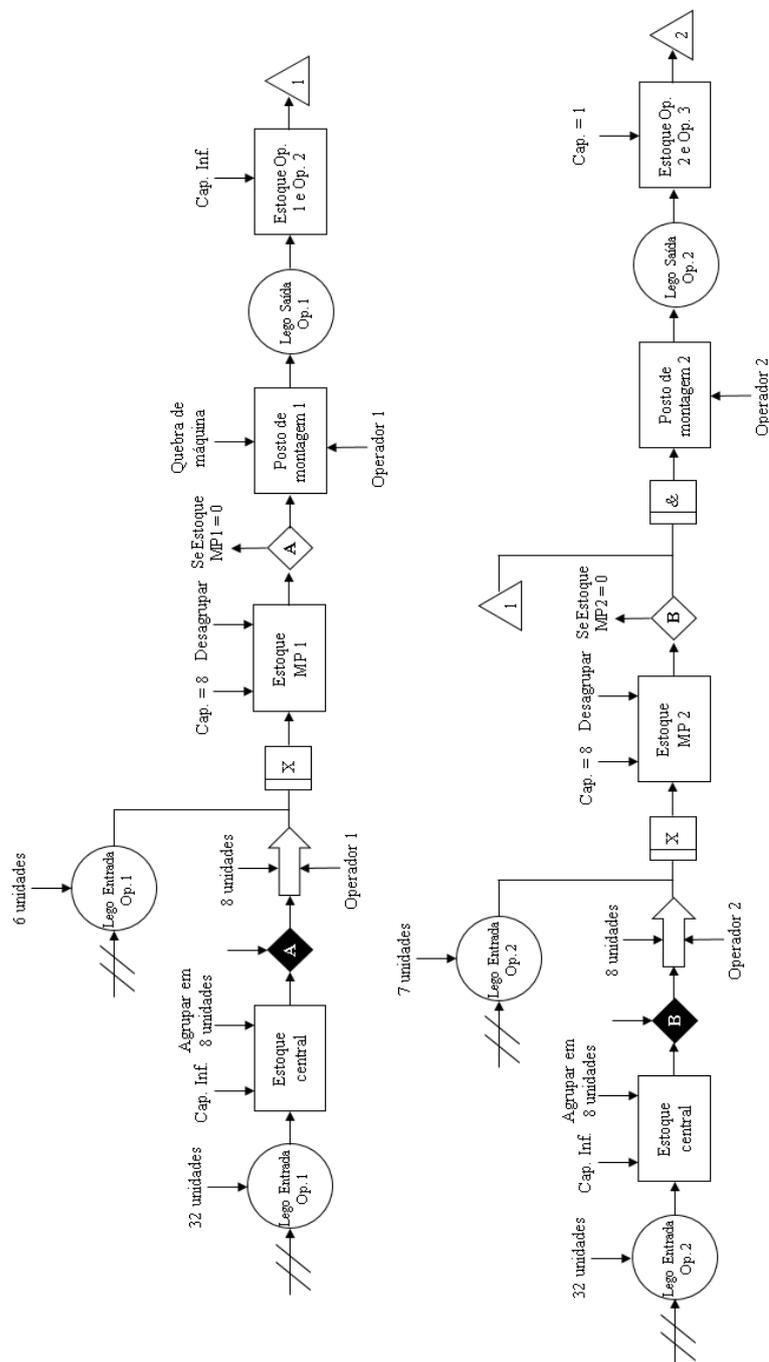
Figura 12 – Linha de montagem de *Karts Lego*[®]

Fonte: Autor.

Ainda segundo a autora, a produção começa no posto 1, que é empurrado. No posto 2, o fluxo é puxado, ou seja, o operador começa a produzir quando o item em seu estoque intermediário é puxado pelo operador seguinte. A etapa seguinte, a do posto 3, produz em lotes de duas unidades. Neste posto há um *kanban* que sinaliza ao operador do posto 4 que um lote foi produzido. Quando o *kanban* é sinalizado, o operador do posto 4 se desloca até o posto 3, recolhe as duas unidades e transporta até seu posto. O operador 4 então finaliza o produto em lotes de quatro unidades. A cada um destes lotes, o operador do posto 5 faz a inspeção, verificando a qualidade dos karts. Os karts aprovados são embalados e conduzidos em lotes de 4 unidades até a área de produtos aceitos. Se o kart for reprovado, o operador 5 o leva até a área de rejeitados.

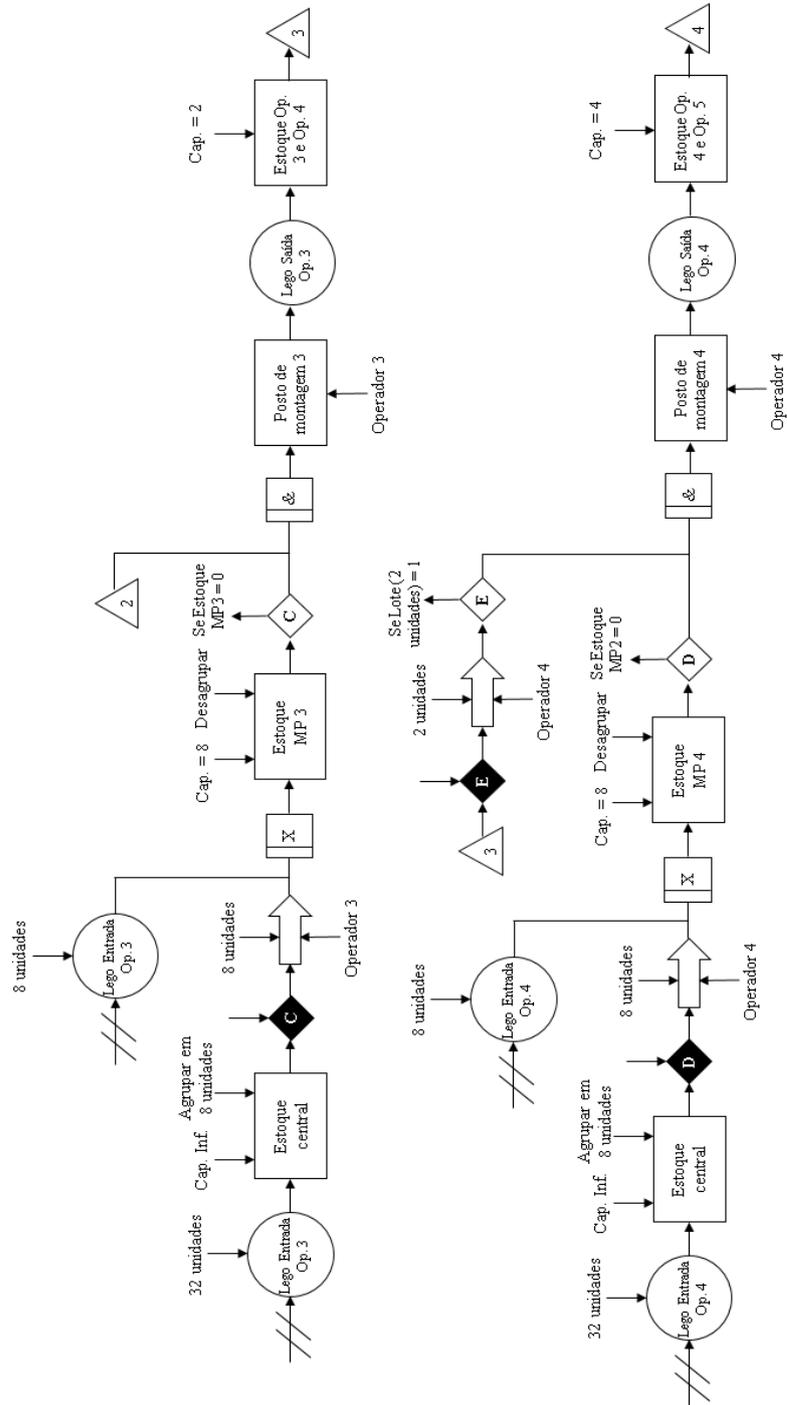
A Figura 12 apresenta como esses postos foram arranjados. O modelo conceitual dessa linha encontra-se representado nas Figura 13, Figura 14 e Figura 15 em formato IDEF-SIM.

Figura 13 – IDEF-SIM da linha de montagem de *Karts Lego*® página 1 de 3

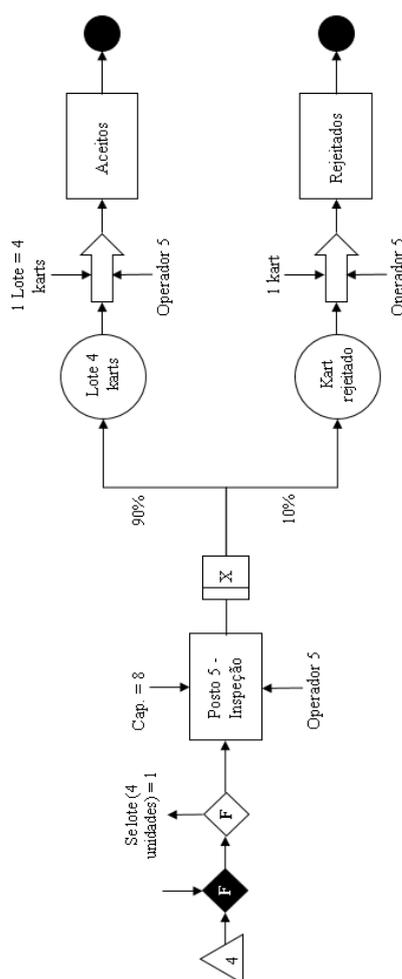


Fonte: Rocha (2020)

Figura 14 – IDEF-SIM da linha de montagem de *Karts Lego*® página 2 de 3



Fonte: Rocha (2020)

Figura 15 – IDEF-SIM da linha de montagem de *Karts Lego*[®] página 3 de 3

Fonte: Rocha (2020)

As filmagens obtidas por Rocha (2020) foram aproveitadas para esta pesquisa. Portanto, não foi necessário realizar novas filmagens para fins deste trabalho. Na subseção seguinte é descrito o processo adotado para extrair os quadros destas imagens.

4.1.2 Definição da taxa e extração dos quadros

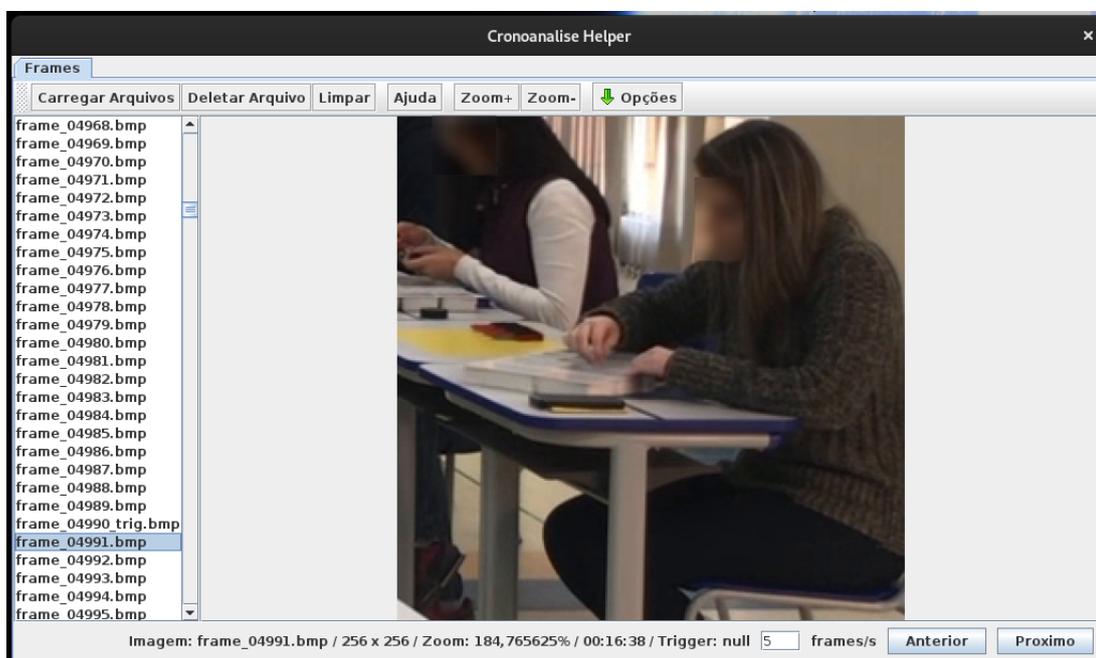
Para que as expressões possam ser identificadas pelo modelo EMOTIC é necessário extrair quadros igualmente espaçados de cada gravação da amostra. Além disso, é preciso que a transição de um quadro para outro permita identificar com clareza os eventos que serão cronometrados durante a cronoanálise dos tempos de operação. Inicialmente foi estabelecido o intervalo de um segundo entre cada quadro para extração. Através de uma validação amostral,

porém, verificou-se que os quadros extraídos a essa taxa não permitiam uma identificação precisa do quadro em que uma operação da linha de montagem inicia e finaliza. Foi então estabelecido um intervalo de 200 milissegundos (0,2s) por quadro para a extração. Nesta taxa, a validação amostral apresentou resultado satisfatório e a taxa foi adotada. A extração foi feita através da ferramenta FFMpeg. Cada quadro extraído foi armazenado em formato em um arquivo *Bitmap* enumerado a partir do valor "00001". Os comandos utilizados para executar a extração estão disponíveis no Anexo B (Comandos FFMPEG).

4.1.3 Cronoanálise dos quadros

Devido a dificuldades em associar os tempos coletados na pesquisa realizada por Rocha (2020) a seus respectivos quadros nas gravações, foi necessário coletá-los novamente. Para marcar os quadros em que os eventos ocorreram, foi feita uma adaptação do *Software* de Análise de Resultados desenvolvido originalmente para validar imagens coletadas por um *Web Crawler* (PAES, 2012). A Figura 16 mostra o *software* em operação.

Figura 16 – *Software* adaptado



Fonte: Autor.

Após carregar o diretório onde foram extraídos os quadros, o *software* exibe a lista de quadros ordenados no canto esquerdo da tela. O quadro selecionado é exibido no centro. Os botões direcionais do teclado permitem navegar entre os quadros. O usuário pode pressionar um

dos botões de zero a nove para registrar um evento. Os eventos registrados são adicionados como sufixo no nome do arquivo do quadro. Estabeleceram-se os seguintes sufixos para identificar eventos nos quadros:

- Sufixo "1" para identificar que uma operação foi iniciada.
- Sufixo "2" para identificar que uma operação foi finalizada.
- Sufixo "3" para identificar que uma operação foi finalizada e uma nova iniciada imediatamente.
- Sufixo "4" para identificar que a operação foi interrompida antes de ser finalizada.
- Sufixo "5" para identificar que uma operação interrompida foi retomada.

Não foi possível coletar todos os eventos em alguns dos *takes*. Isso se deve a obstruções nas imagens que impedem a identificação com clareza do momento em que o evento ocorreu. Nesses casos, foi descartada a operação que esse evento faria parte para fins de cronoanálise. A Tabela 3 apresenta os trechos que precisaram ser descartados.

Tabela 3 – Trechos desconsiderados na cronoanálise.

<i>Take</i>	Operador	Primeiro quadro	Último quadro	Início	Final
16	4	21757	21851	1:12:31	1:12:50
16	4	26221	26368	1:27:24	1:27:54
16	4	26543	31763	1:28:29	1:45:53
16	5	21757	21851	1:12:31	1:12:50
16	5	26221	26368	1:27:24	1:27:54
16	5	26543	31763	1:28:29	1:45:53
23	2	17118	17617	0:57:04	0:58:43
23	3	17118	17617	0:57:04	0:58:43

Fonte: Autor

Após a coleta em todos os vídeos disponíveis, os eventos foram demarcados através do *software*, tabulados em uma planilha eletrônica e exportados no formato CSV (valores separados por vírgula). A Tabela 4 apresenta o total de ocorrências de cada evento para cada operador durante a cronoanálise.

Tabela 4 – Totais da cronoanálise.

Operador	Evento	Total de Ocorrências
1	1	66
1	2	66
1	3	401
1	4	20
1	5	20
1	operação completa	573
2	1	468
2	2	468
2	3	0
2	4	0
2	5	0
2	operação completa	936
3	1	61
3	2	61
3	3	415
3	4	0
3	5	0
3	operação completa	537
4	1	237
4	2	237
4	3	235
4	4	0
4	5	0
4	operação completa	709
5	1	206
5	2	206
5	3	258
5	4	0
5	5	0
5	operação completa	670

Fonte: Autor

4.1.4 Detecção dos indivíduos

O modelo EMOTIC precisa extrair um retângulo contendo o indivíduo a ser analisado do contexto da imagem. No entanto, não é disponibilizada uma maneira de delimitar esse retângulo. Este precisa ser fornecido junto à imagem para que o modelo seja executado corretamente.

Devido ao elevado número de quadros, a marcação de retângulos manual é inviável. Optou-se então por utilizar um segundo modelo de aprendizagem de máquina para fazer essa demarcação. O modelo escolhido foi o "*Object detection model trained on Open Images V4 with ImageNet pre-trained Inception Resnet V2 as image feature extractor*" (TFHUB, 2021). Esse modelo identifica e demarca classes de objetos em uma imagem, incluindo "pessoa", "homem" e "mulher".

O modelo foi executado quadro a quadro para detectar os indivíduos. Em cada quadro foram selecionados os retângulos representando a categoria "pessoa" e armazenados. No total 4.885.245 pessoas foram detectadas. Os resultados foram armazenados para serem utilizados na detecção de expressões. A Figura 17 apresenta um exemplo de quadro com as pessoas detectadas.

Figura 17 – Pessoas detectadas



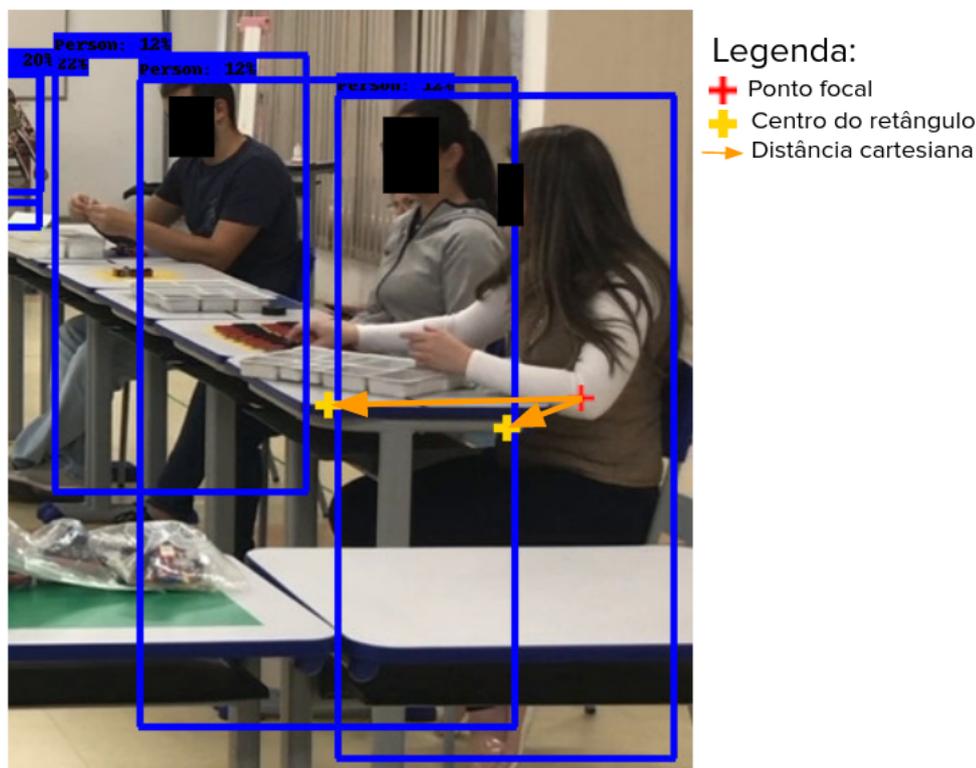
Fonte: Autor.

4.1.5 Identificação dos operadores

Na Figura 17 observa-se que o modelo detecta alguns falso positivos. Além disso, nem todos os indivíduos presentes na gravação são operadores da linha de produção. É necessário identificar qual retângulo corresponde a qual operador para que seja possível determinar seu humor aparente.

A identificação dos operadores foi feita a partir de um ponto central, que foi representado na Figura 18. Este ponto foi determinado manualmente para cada posto de trabalho. Cada ponto central foi posicionado próximo ao centro de onde o operador foi localizado. Estes valores foram tabulados e armazenados em formato CSV (valores separados por vírgula). Em cada quadro foi calculada a distância cartesiana entre o ponto central do posto de trabalho e o centro de cada retângulo. A Equação 4.1 mostra como essa distância (d) é calculada, onde x_c e x_r são as posições do ponto central e do centro do retângulo no eixo horizontal respectivamente e y_c e y_r as posições no eixo vertical. Definiu-se então que um retângulo que representa um determinado operador deve conter o ponto focal determinado para o operador no vídeo. Caso mais de um

Figura 18 – Ponto central



Fonte: Autor.

ponto satisfaça a essa condição, o retângulo com a menor distância cartesiana de seu centro até o ponto central é selecionado para representar o operador.

$$d = \sqrt{(x_c - x_r)^2 + (y_c - y_r)^2} \quad (4.1)$$

Para cada operador foi selecionada uma amostra de quadros para validação. Em seguida, foi executado um *script* que desenha somente o retângulo selecionado em cada quadro. A Figura 19 apresenta um exemplo de quadro para ser validado.

Os quadros foram validados manualmente. Quando o resultado verificado não era satisfatório, o ponto focal foi editado e o processo repetido. Quando satisfatório, os retângulos foram definidos para aquele operador. Os *scripts* utilizados para identificar os pontos centrais e para desenhar os retângulos para a validação estão disponíveis nos Anexo C (*Script* para determinar pontos centrais) e Anexo E (*Script* para validação do ponto central) respectivamente. Os retângulos selecionados para cada operador foram armazenados em formato JSON para serem utilizados nos processamentos seguintes.

Figura 19 – Validação do ponto central



Fonte: Autor.

É importante salientar que durante este processo, o operador 2 foi o que mais apresentou erros no processo de validação devido à posição entre os operadores 1 e 3 que pode ser observada na Figura 19. Embora tenha sido possível estabelecer um ponto central satisfatório para este operador, é esperado que este apresente um número maior de erros de identificação quando comparado aos demais.

4.2 Identificação das emoções

4.2.1 Classificação do humor aparente

De posse das imagens dos quadros e das posições e dimensões dos retângulos, foi executado o modelo EMOTIC. Para tal foi utilizada uma implementação pré treinada do modelo baseada no *framework* Pytorch. Os arquivos fontes, bem como os valores de limiares do modelo de *Deep Learning* foram baixados, configurados e executados em cada quadro para cada operador.

No final desse procedimento, as três variáveis discretas foram calculadas para cada um dos 4.885.245 retângulos e 17.512.534 classes discretas foram detectadas. Esses resultados foram então cruzados aos dados de cronoanálise.

O resultado produzido pela implementação do modelo EMOTIC utilizada é um arquivo

de texto que em cada linha ficam inseridos em sequência o nome do arquivo, os pontos que definem o retângulo analisado, o valor aferido para cada uma das três variáveis contínuas e o nome das variáveis discretas detectadas. Este arquivo foi armazenado para ser utilizado nos processamentos seguintes.

4.2.2 Cruzamento de dados

Nesse ponto todos os dados utilizados para o estudo já estão disponíveis, porém de maneira isolada não trazem qualquer informação útil para a análise. É necessário associar os diversos resultados obtidos em diferentes etapas para que seja possível analisá-los. Na Figura 11 esta etapa é representada pelo item "7. Cruzamento de Dados". Nesse processo, os eventos de cronoanálise, os retângulos associados a cada operador em cada instante das gravações e o resultado da classificação do humor aparente do modelo EMOTIC são associados para obter os tempos de cada operação associado ao valor detectado para cada variável de humor. Os processamentos descritos a seguir foram realizados utilizando a ferramenta Apache Spark™. O código fonte dos *scripts* utilizados está disponível no Anexo D (*Script* para cruzamento de dados).

4.2.2.1 Identificação das iterações dos processos e cálculo da duração

Na etapa de cronoanálise, descrita na subseção 4.1.3, foi produzido um arquivo em formato CSV contendo os eventos observados na etapa associados a seus respectivos quadros. Esse arquivo foi importado e dele foram deduzidas as seguintes informações:

1. Os quadros que fazem parte de cada iteração do processo de produção para cada operador em seu respectivo turno.
2. Através do quadro inicial e final de cada iteração foi extrapolada sua duração. Sendo ela $(q_f - q_i)/5$ em segundos. Onde q_i é o número do quadro inicial e q_f o número do quadro final.

Durante a identificação das iterações dos processos (item 1) algumas inconsistências foram observadas. Essas inconsistências ocorrem quando um operador encerra ou interrompe uma operação sem tê-la iniciado ou quando inicia uma nova operação sem finalizar a anterior. A

causa para esses casos foi atribuída a falhas humanas durante a cronoanálise. Cada um desses casos foi analisado frente aos respectivos quadros e as inconsistências corrigidas.

Além das inconsistências, também foi feita uma conferência frente aos quadros dos *outliers* de tempo de operação (item 2). Os erros constatados foram corrigidos e o cruzamento refeito. Esses procedimentos de conferência foram repetidos até que nenhum erro fosse encontrado.

4.2.2.2 Cálculo da intensidade das variáveis de humor aparente dentro da iteração do processo

Os retângulos selecionados para cada operador em cada quadro através dos procedimentos descritos na subseção 4.1.5 (Identificação dos operadores) foram selecionados dentro dos resultados obtidos na etapa descrita pela subseção 4.2.1 (Classificação do humor aparente). Desta forma foi possível associar o operador aos valores de humor aparente aferidos para ele em cada quadro. Dentro desse conjunto de dados foram agrupados por operação e a média das variáveis de humor aparente foi calculada. Para o cálculo das médias das variáveis discretas foi considerado o valor 1 quando o quadro contém a variável e zero do contrário.

Feitas as transformações com o Apache Spark™ foi exportada uma coleção de dados contendo o resultado agregado das expressões para cada iteração da operação em formato CSV. O próximo capítulo detalha como esses dados foram utilizados para elaborar um modelo de simulação.

4.3 Elaboração do modelo

4.3.1 Correlação do humor com o tempo de processamento

Para determinar se há correlação entre as variáveis calculadas pelo EMOTIC e os tempos de processamento foi calculado o Coeficiente de Correlação de Spearman. Neste cálculo são produzidos dois valores: o coeficiente de correlação (r_s) e o *P-Value* (P_s) (BAKHTIAR *et al.*, 2021; REBEKIĆ *et al.*, 2015). O valor de r_s pode variar entre -1 e 1. Onde valores negativos indicam correlação negativa e valores positivos, correlação positiva. Quanto mais distante de zero o valor deste coeficiente, mais forte a correlação. O valor de (P_s) pode variar entre 0 e 1. No Coeficiente de Spearman, a hipótese nula (H_0) representa que não há correlação entre as variáveis. Enquanto a hipótese alternativa (H_1) representa que há correlação entre as variáveis. Portanto,

quanto menor o valor de P_s , ou seja, mais próximo a zero, maior a significância estatística da correlação.

Foi calculada uma matriz de correlação para todos os operadores e para cada operador. Esta matriz possui duas dimensões contendo cada variável disponível nos dados, incluindo tempo de processamento e cada uma das 29 variáveis do modelo EMOTIC. O cálculo foi feito em linguagem R. Foram então extraídas das matrizes as correlações entre as variáveis do modelo EMOTIC e o tempo de operação. Estes valores foram tabulados no Anexo A. A Tabela 5 abaixo apresenta os valores mais significantes encontrados para cada operador.

Tabela 5 – Correlação da emoção aparente com a duração do processo.

Operador	Tipo da Variável	Variável	Correlação	<i>P-Value</i>	Ocorrências
Todos	Discreta	Desconexão	-0,40005	0	1.499.025
Todos	Discreta	Felicidade	0,29942	0	10.886
Todos	Discreta	Antecipação	0,29253	0	148.144
1	Discreta	Antecipação	-0,19524	$1,344 \times 10^{-05}$	21.305
1	Contínua	Excitação	0,16627	$2,182 \times 10^{-04}$	245.327
1	Contínua	Domínio	-0,16554	$2,328 \times 10^{-04}$	245.327
2	Discreta	Dúvida / Confusão	-0,13306	$3,931 \times 10^{-03}$	352.953
2	Contínua	Valência	0,09929	$3,175 \times 10^{-02}$	386.469
2	Discreta	Vergonha	0,09537	$3,917 \times 10^{-02}$	694
3	Contínua	Domínio	0,25696	$1,291 \times 10^{-08}$	372.282
3	Discreta	Inquietação	-0,20482	$6,649 \times 10^{-06}$	216.434
3	Contínua	Valência	0,1745	$1,300 \times 10^{-04}$	372.282
4	Contínua	Domínio	0,16881	$2,293 \times 10^{-04}$	385.903
4	Discreta	Paz	0,12137	$8,300 \times 10^{-03}$	3.762
4	Discreta	Desconexão	-0,11489	$1,250 \times 10^{-02}$	296.266
5	Discreta	Antecipação	0,33231	$1,883 \times 10^{-13}$	22.055
5	Discreta	Desconexão	-0,30163	$3,091 \times 10^{-11}$	356.121
5	Contínua	Domínio	0,28036	$7,589 \times 10^{-10}$	388.589

Fonte: Autor

A Tabela 5 apresenta o valor da correlação (r_s), o *P-Value* (P_s) e o número de ocorrências dessa variável na amostra, ou seja, quantas vezes ela foi observada nos quadros processados, para cada uma das 29 variáveis no geral e também para cada um dos quatro operadores da linha. Um total de 66 correlações significantes ($P_s \leq 5\%$) foi encontrado. Portanto, a hipótese 1 definida

na seção 1.1 foi confirmada: vários dos atributos de humor identificados possuem correlação estatística com o tempo de operação.

Além disso, é possível verificar que a correlação com a variável Domínio é significativa para quase todos os operadores. Isto sugere que a correlação de Domínio com o tempo de operação não depende do indivíduo nem da tarefa específica que este indivíduo executa.

Também na Tabela 5, é possível observar que alguns *P-Values* possuem valor zero. Isso é teoricamente impossível. O valor foi observado porque os valores calculados eram de uma escala pequena demais para o interpretador da linguagem R e precisaram ser arredondados para zero.

4.3.2 Seleção de variáveis

É possível identificar na Tabela 5 um total de 66 correlações com *P-Value* inferior a 5%. Ou seja, há 66 correlações com elevado grau de certeza dentro da matriz de Spearman calculada. Sendo elas:

- 22 para quando se considera todos os operadores;
- 8 para o operador 1;
- 5 para o operador 2;
- 14 para o operador 3;
- 4 para o operador 4;
- 13 para o operador 5.

Devido ao elevado número de variáveis correlatas ao tempo de operação, optou-se por selecionar uma variável para cada operador segundo os seguintes critérios:

1. a variável possui *P-Value* inferior a 5% para o operador;
2. caso mais de uma variável se enquadre, são selecionadas as variáveis com menor valor de *P-Value*;
3. caso haja empate no quesito anterior, é selecionada a variável com maior valor absoluto para a correlação.

A Tabela 6 apresenta as variáveis selecionadas para cada operador extraídas da Tabela 5. O código fonte dos *scripts* em linguagem R utilizados nesta seção estão disponíveis no Anexo F (*Scripts* para cálculo da correlação e seleção de variáveis).

Tabela 6 – Variáveis selecionadas para cada operador.

Operador	Tipo da Variável	Variável	Correlação	P-Value	Ocorrências
1	Discreta	Antecipação	-0,19524	1,344 ⁻⁰⁵	21.305
2	Discreta	Dúvida / Confusão	-0,13306	3,931 ⁻⁰³	352.953
3	Contínua	Domínio	0,25696	1,291 ⁻⁰⁸	372.282
4	Contínua	Domínio	0,16881	2,293 ⁻⁰⁴	385.903
5	Discreta	Antecipação	0,33231	1,883 ⁻¹³	22.055

Fonte: Autor

4.3.3 Definição das classes de humor

Para cada operador foram determinadas quatro faixas de valor para que fosse possível discretizar a variável contínua que representa a variável de humor. Os limites de cada faixa foram calculados para coincidir com os quartis dos valores aferidos para a variável selecionando os valores referentes ao operador. As condições de contorno abaixo foram aplicadas para alguns casos encontrados:

- quando uma classe possuía um único valor, esse valor foi transferido das outras classes para esta classe;
- quando duas ou mais classes possuíam um mesmo valor único, essas classes foram unidas para formar uma única classe.

Por exemplo: em uma amostra contendo 30 valores iguais a 0 e 70 valores distintos maiores que 0, o primeiro quartil seria composto por 25 valores 0 e o segundo quartil iria conter outros 5 valores 0. Aplicando a primeira condição de contorno acima, os 5 valores iguais a 0 são transferidos do segundo grupo para o primeiro resultando em um grupo de 30 valores iguais a 0, um segundo grupo de 20 valores distintos e outros dois grupos de 25 valores distintos.

Em um segundo exemplo: uma amostra contendo 60 valores iguais a 0 e 40 valores distintos maiores que 0, seguindo a divisão dos quartis, seria formado por dois grupos contendo 25 valores iguais a 0 cada, um terceiro grupo contendo 10 valores iguais a 0 e 15 valores distintos

e um último grupo contendo 25 valores distintos. Aplicando a primeira condição de contorno seguida pela segunda, temos apenas três grupos: um grupo composto por 60 valores 0, um composto por 15 valores distintos e um terceiro e último grupo formado por outros 25 valores distintos.

A partir dessas faixas foram definidas quatro classes para classificar cada iteração do processo de produção executado pelo operador em questão. Na Tabela 7 são apresentadas as faixas definidas para cada classe.

Tabela 7 – Classes de humor por operador.

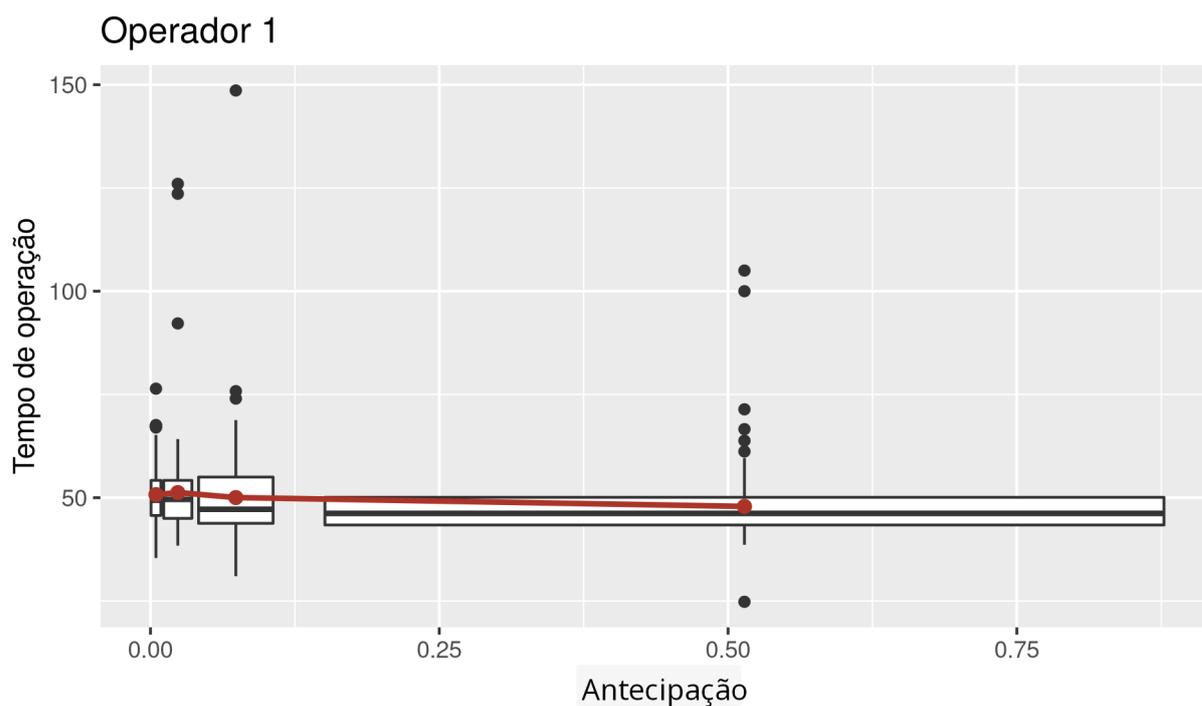
Operador	Variável	Valor Inicial	Valor Final	Percentual Inicial	Percentual Final	Tamanho do conjunto
1	Antecipação	0,0000	0,0107	0 %	25 %	126
1	Antecipação	0,0107	0,0385	25 %	50 %	125
1	Antecipação	0,0385	0,1110	50 %	75 %	126
1	Antecipação	0,1110	0,9177	75 %	100 %	126
2	Dúvida / Confusão	0,4533	0,9000	0 %	25 %	120
2	Dúvida / Confusão	0,9000	0,9689	25 %	50 %	120
2	Dúvida / Confusão	0,9689	0,9937	50 %	75 %	119
2	Dúvida / Confusão	0,9937	1,0000	75 %	100 %	122
3	Domínio	5,5278	5,8022	0 %	25 %	122
3	Domínio	5,8022	5,8784	25 %	50 %	122
3	Domínio	5,8784	5,9571	50 %	75 %	122
3	Domínio	5,9571	6,1698	75 %	100 %	123
4	Domínio	5,4783	6,0576	0 %	25 %	121
4	Domínio	6,0576	6,1720	25 %	50 %	121
4	Domínio	6,1720	6,2849	50 %	75 %	121
4	Domínio	6,2849	6,6394	75 %	100 %	122
5	Antecipação	0,0000	0,0057	0 %	25 %	120
5	Antecipação	0,0057	0,0223	25 %	50 %	119
5	Antecipação	0,0223	0,0680	50 %	75 %	119
5	Antecipação	0,0680	0,6087	75 %	100 %	120

Fonte: Autor

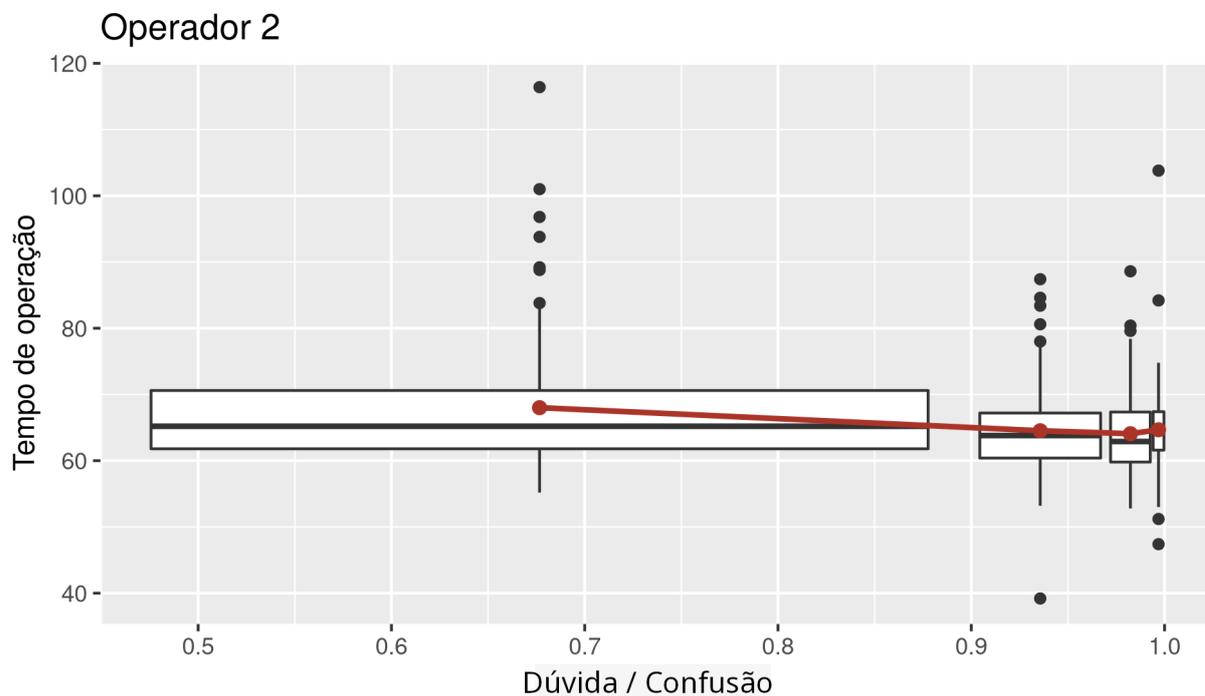
As tendências para o tempo de operação nas diferentes classes definidas acima podem ser observadas graficamente através de *boxplots* como os das figuras a seguir. Na Figura 20 observa-se uma tendência de queda no tempo de operação do operador 1 à medida que a variável Antecipação aumenta. Essa tendência é observada tanto na média quanto nos limites dos quartis. Uma tendência similar é observada na Figura 21 para o operador 2 e a variável Dúvida / Confusão. Já nas figuras Figura 22, Figura 23 e Figura 24 uma tendência inversa é observada. À medida que

as variáveis Domínio para os operadores 3 e a Antecipação para o operador 5 respectivamente aumentam, o tempo de operação tende a aumentar. É importante destacar que os *boxplots* mais estreitos representam uma concentração maior de valores em determinada faixa. De maneira análoga, *boxplots* mais largos representam uma concentração menor.

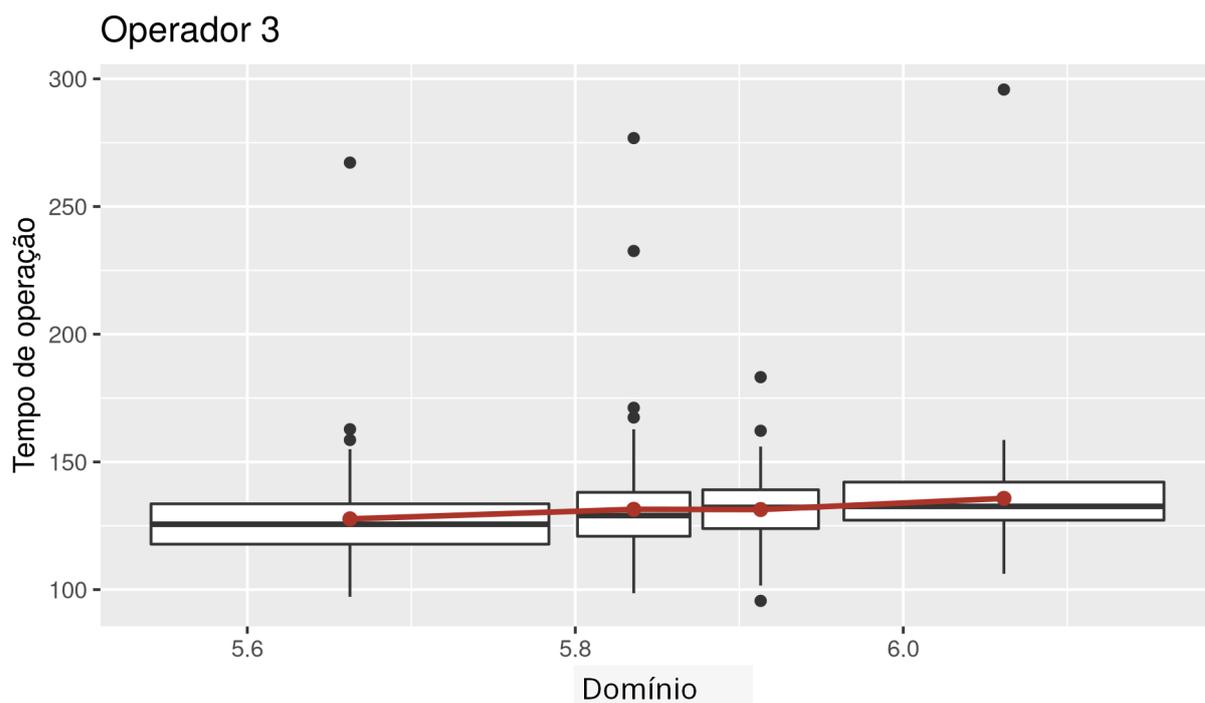
Figura 20 – *Boxplot* de duração do processo por classe para o operador 1



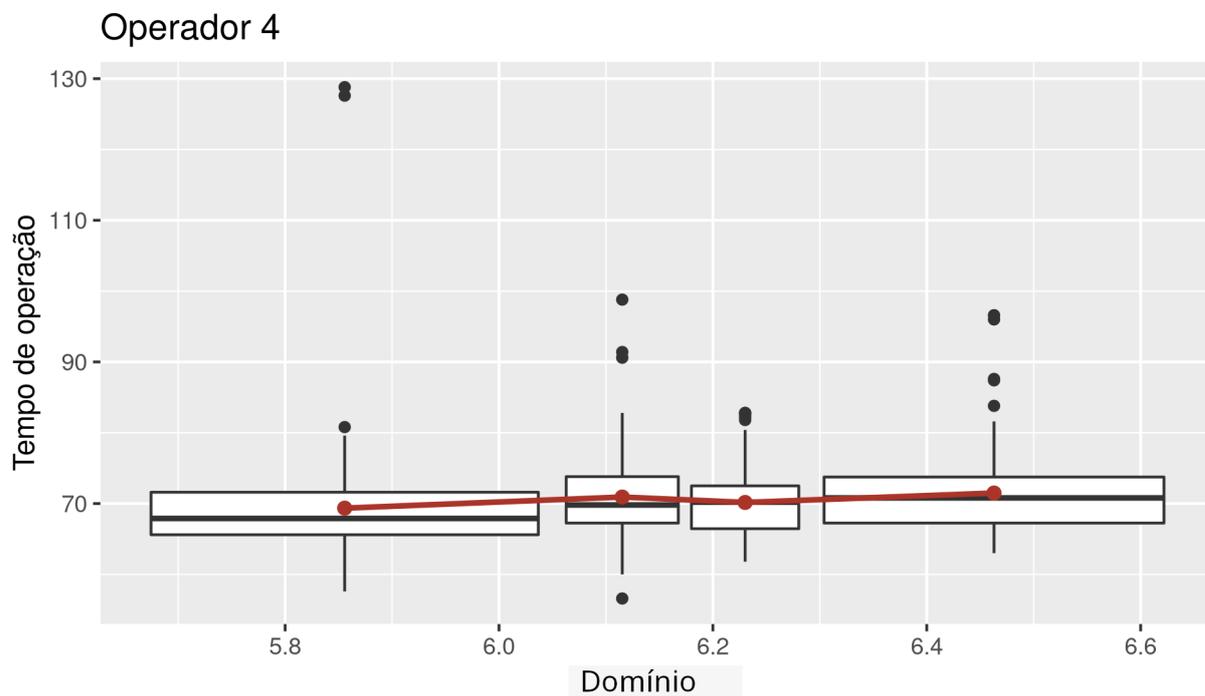
Fonte: Autor.

Figura 21 – *Boxplot* de duração do processo por classe para o operador 2

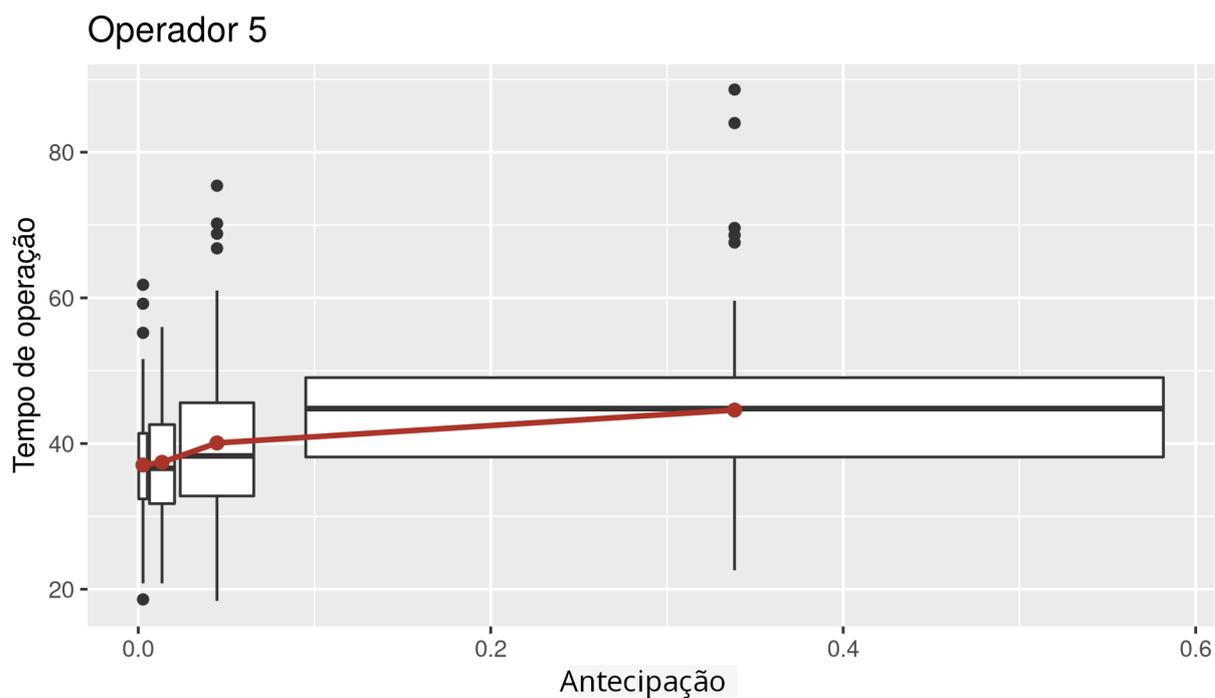
Fonte: Autor.

Figura 22 – *Boxplot* de duração do processo por classe para o operador 3

Fonte: Autor.

Figura 23 – *Boxplot* de duração do processo por classe para o operador 4

Fonte: Autor.

Figura 24 – *Boxplot* de duração do processo por classe para o operador 5

Fonte: Autor.

4.3.4 Cálculo das curvas de distribuição de probabilidade

Uma vez determinadas as diferentes classes de humor que serão modeladas, foi calculado o melhor *fit* estatístico para os tempos como um todo e para cada uma das classes. Dessa forma será possível modelar os tempos de processamento para um cenário base e para cenários em diferentes classes de humor.

Para realizar este processo foi utilizado o *software* ExpertFit, que acompanha o *software* de simulação Flexsim[®]. Um resumo das curvas de distribuição de probabilidade calculadas está disponível na Tabela 8.

Tabela 8 – Classes de humor por operador.

Oper.	Classe	Modelo de distribuição	Score	Parâmetros	Avaliação
1	Base	Log-Logística(E)	100,00%	Localização: 24,09868 Escala: 24,28431 Forma: 5,90994	Intermediária
1	Antecipação 0% a 25%	Pearson Tipo V(E)	97,66%	Localização: 21,87731 Escala: 427,18992 Forma: 15,92927	Boa
1	Antecipação 25% a 50%	Log-Logística(E)	100,00%	Localização: 28,82837 Escala: 20,74329 Forma: 4,99439	Boa
1	Antecipação 50% a 75%	Log-Logística(E)	100,00%	Localização: 28,10705 Escala: 20,12302 Forma: 4,53384	Boa
1	Antecipação 75% a 100%	Log-Logística(E)	98,39%	Localização: 20,65280 Escala: 25,71390 Forma: 7,55950	Intermediária
2	Base	Log-Laplace(E)	99,14%	Localização: 38,65363 Escala: 25,54637 Forma: 5,20113	Boa
2	Dúvida / Confusão 0% a 25%	Log-Logística(E)	100,00%	Localização: 53,42710 Escala: 12,23768 Forma: 2,98857	Boa
2	Dúvida / Confusão 25% a 50%	Log-Laplace(E)	99,19%	Localização: 32,64474 Escala: 31,15526 Forma: 6,54342	Boa

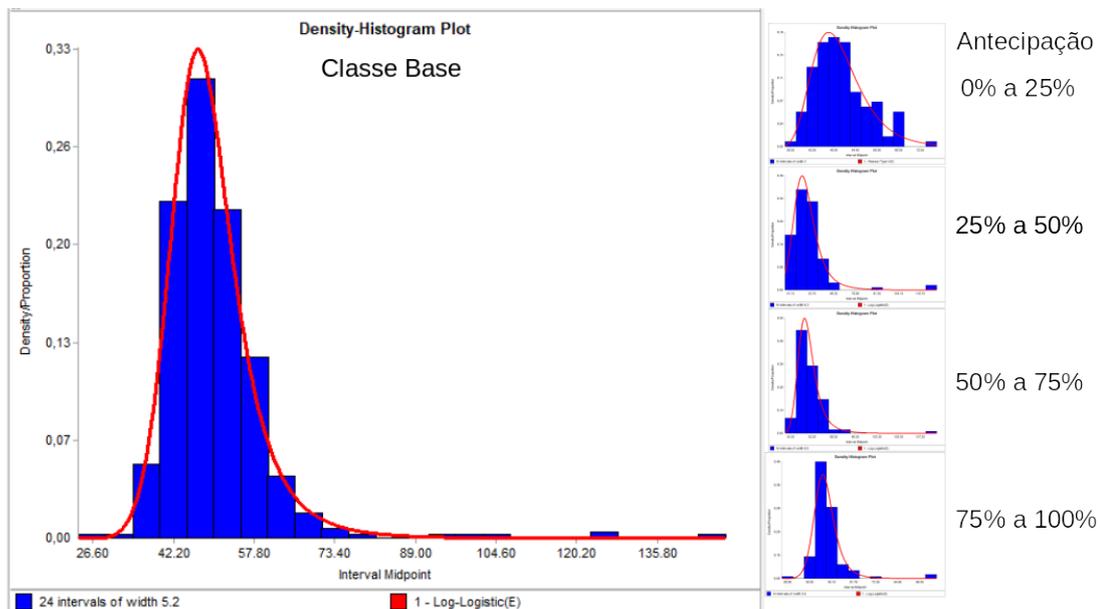
2	Dúvida / Confusão 50% a 75%	Pearson Tipo V(E)	92,74%	Localização: 41,45745 Escala: 293,69520 Forma: 13,99335	Boa
2	Dúvida / Confusão 75% a 100%	Log-Laplace	95,97%	Localização: 0,00000 Escala: 64,49992 Forma: 16,47403	Boa
3	Base	Log-Logística	98,39%	Localização: 0,00000 Escala: 130,06860 Forma: 17,31974	Boa
3	Domínio 0% a 25%	Log-Logística(E)	100,00%	Localização: 91,77864 Escala: 33,44876 Forma: 4,66748	Boa
3	Domínio 25% a 50%	Log-Logística	100,00%	Localização: 0,00000 Escala: 129,07046 Forma: 16,93371	Boa
3	Domínio 50% a 75%	Log-Logística	96,77%	Localização: 0,00000 Escala: 131,21908 Forma: 18,23898	Boa
3	Domínio 75% a 100%	Log-Logística(E)	100,00%	Localização: 87,04355 Escala: 46,65464 Forma: 7,15421	Boa
4	Base	Log-Logística(E)	100,00%	Localização: 48,90856 Escala: 20,73285 Forma: 7,11957	Boa
4	Domínio 0% a 25%	Log-Logística(E)	98,39%	Localização: 43,24256 Escala: 24,95091 Forma: 8,11274	Boa
4	Domínio 25% a 50%	Log-Logística(E)	100,00%	Localização: 52,23838 Escala: 17,87664 Forma: 5,96975	Boa
4	Domínio 50% a 75%	Erlang(E)	88,71%	Localização: 55,74254 Escala: 1,59866 Forma: 9	Boa
4	Domínio 75% a 100%	Pearson Tipo V(E)	94,35%	Localização: 52,85606 Escala: 235,13436 Forma: 13,65346	Boa

5	Base	Log-Logística	91,67%	Localização: 0,00000 Escala: 38,71886 Forma: 7,43992	Boa
5	Antecipação 0% a 25%	Log-Logística	99,17%	Localização: 0,00000 Escala: 36,61401 Forma: 9,10788	Boa
5	Antecipação 25% a 50%	Log-Logística(E)	86,29%	Localização: 7,66152 Escala: 28,60874 Forma: 6,06196	Boa
5	Antecipação 50% a 75%	Pearson Tipo VI	93,33%	Localização: 0,00000 Escala: 25,69391 Forma #1: 40,32708 Forma #2: 26,66023	Boa
5	Antecipação 75% a 100%	Log-Logística	96,77%	Localização: 0,00000 Escala: 43,71970 Forma: 7,96841	Boa

Fonte: Autor

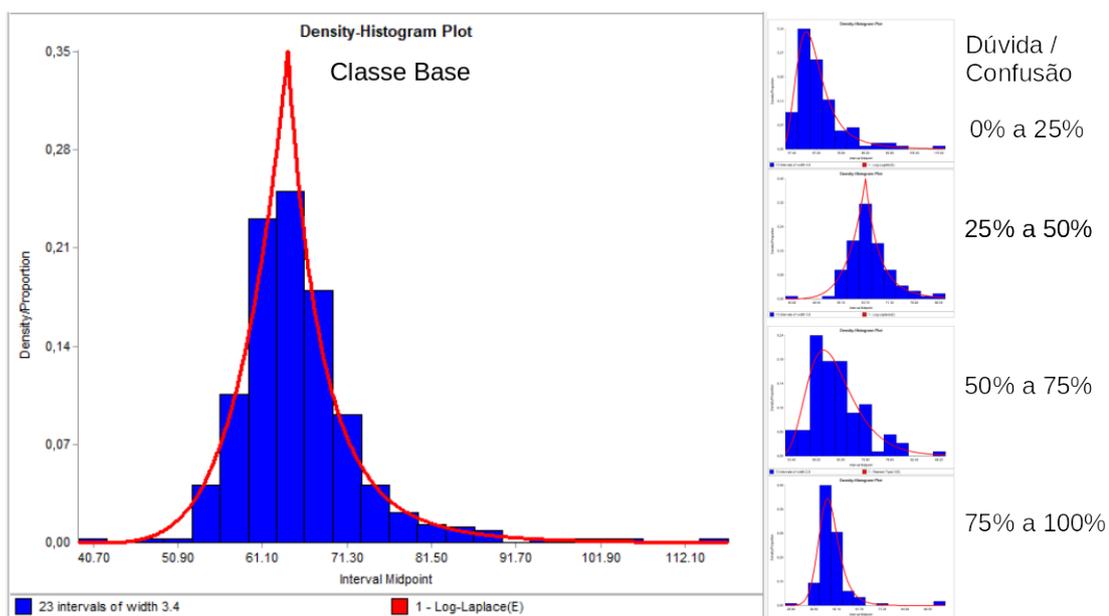
A qualidade do *fit* estatístico também pode ser verificada graficamente nos histogramas da Figura 25 até a Figura 29 resumidamente. Os histogramas completos estão disponíveis no Anexo C.

Figura 25 – Histogramas para o operador 1



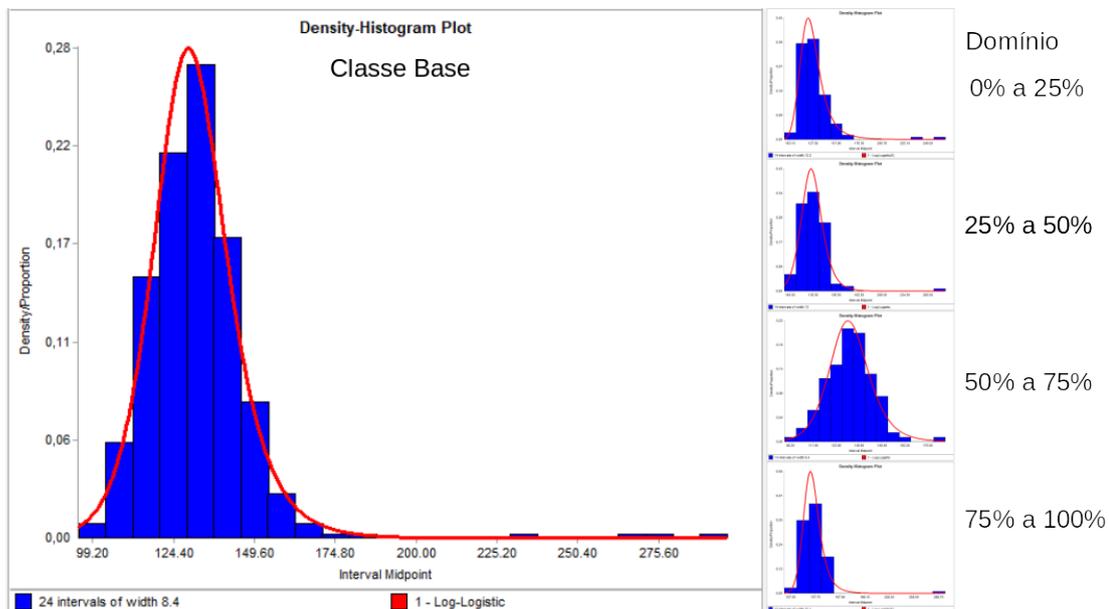
Fonte: Autor.

Figura 26 – Histogramas para o operador 2



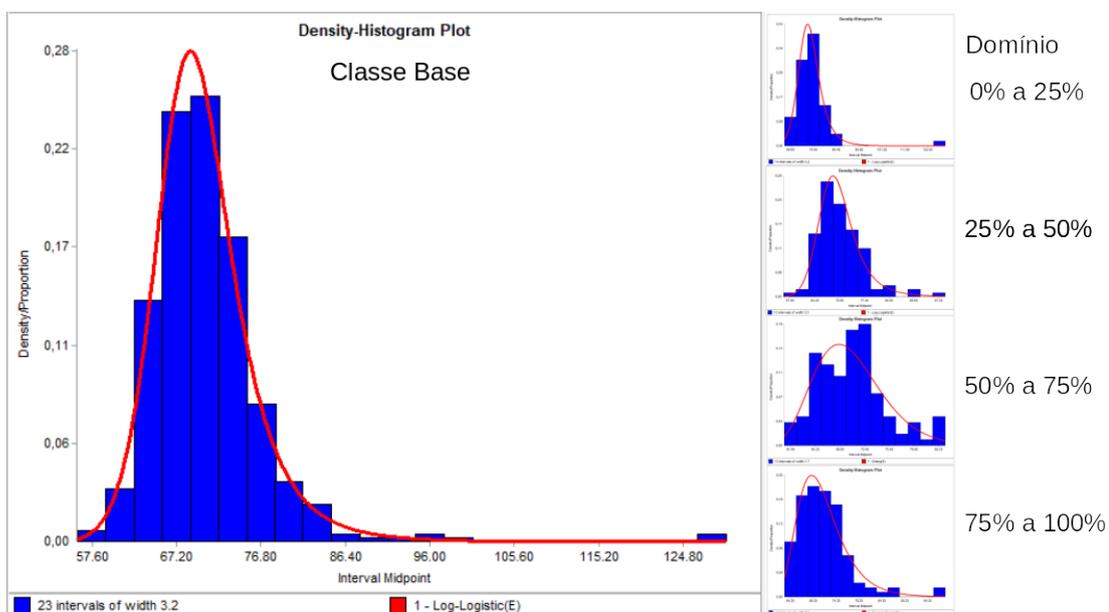
Fonte: Autor.

Figura 27 – Histogramas para o operador 3



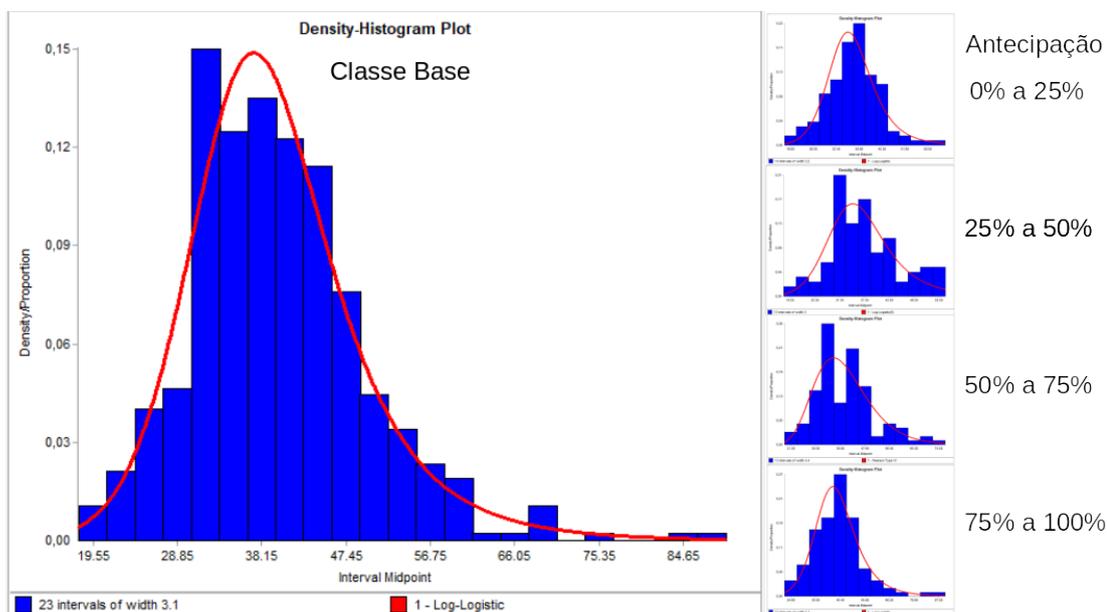
Fonte: Autor.

Figura 28 – Histogramas para o operador 4



Fonte: Autor.

Figura 29 – Histogramas para o operador 5



Fonte: Autor.

Observando os histogramas e a Tabela 8, é possível identificar variações significativas nas curvas de distribuição de probabilidade para cada classe estabelecida. Portanto, a hipótese 2 estabelecida na seção 1.1 foi confirmada: a curva de distribuição de probabilidade dos tempos de processamento é diferente quando as amostras consideram classes diferentes de humor para um operador.

4.4 Validação do modelo

No trabalho de Rocha (2020) foi criado um modelo de simulação a eventos discretos utilizando o *software* Flexsim[®]. O modelo de simulação implementa o modelo conceitual proposto. Esse modelo será adaptado e utilizado para validação dos resultados desta pesquisa. A Figura 30 apresenta a representação gráfica do modelo de simulação construído.

Figura 30 – Modelo de simulação



Fonte: Rocha (2020)

Este modelo foi aproveitado e modificado para a realização do experimento. Embora o modelo em si tenha sido aproveitado, os tempos de operação de todos os operadores foram ajustados de acordo com as estatísticas de cada cenário proposto para este experimento. As subseções a seguir discorrem sobre como isto foi feito.

4.4.1 Preparação dos cenários

Este trabalho utilizou 12 filmagens disponibilizadas por Rocha (2020). Cada filmagem representa um dia distinto que a linha foi colocada em operação até que produzisse um total de 40 carrinhos. Cada uma dessas filmagens foi designada para representar um turno, de 1 a 12.

Foram preparados dois cenários para verificar os efeitos do método proposto nesta pesquisa no modelo de simulação disponibilizado por Rocha (2020). O primeiro é o Cenário Base e utiliza as curvas de distribuição de probabilidade da Classe Base de cada operador. O segundo é o Cenário Proposto. Neste cenário foi calculada uma Classe de humor para cada operador em cada turno. A classe designada para cada turno foi aquela cuja faixa de valores contém a média para a variável de humor do operador. A Tabela 9 apresenta as classes selecionadas para cada operador em cada turno.

Tabela 9 – Cenário Proposto.

Operador	Turno	Classe
1	1	Antecipação de 0% a 25%
1	2	Antecipação de 50% a 75%
1	3	Antecipação de 50% a 75%
1	4	Antecipação de 75% a 100%
1	5	Antecipação de 50% a 75%
1	6	Antecipação de 25% a 50%
1	7	Antecipação de 50% a 75%
1	8	Antecipação de 75% a 100%
1	9	Antecipação de 50% a 75%
1	10	Antecipação de 50% a 75%
1	11	Antecipação de 50% a 75%
1	12	Antecipação de 75% a 100%
2	1	Dúvida / Confusão de 50% a 75%
2	2	Dúvida / Confusão de 50% a 75%
2	3	Dúvida / Confusão de 0% a 25%
2	4	Dúvida / Confusão de 25% a 50%
2	5	Dúvida / Confusão de 25% a 50%
2	6	Dúvida / Confusão de 50% a 75%
2	7	Dúvida / Confusão de 25% a 50%
2	8	Dúvida / Confusão de 50% a 75%
2	9	Dúvida / Confusão de 0% a 25%
2	10	Dúvida / Confusão de 25% a 50%
2	11	Dúvida / Confusão de 0% a 25%
2	12	Dúvida / Confusão de 25% a 50%
3	1	Domínio de 25% a 50%
3	2	Domínio de 25% a 50%
3	3	Domínio de 0% a 25%
3	4	Domínio de 25% a 50%
3	5	Domínio de 50% a 75%
3	6	Domínio de 50% a 75%
3	7	Domínio de 25% a 50%
3	8	Domínio de 50% a 75%
3	9	Domínio de 75% a 100%
3	10	Domínio de 50% a 75%
3	11	Domínio de 50% a 75%
3	12	Domínio de 25% a 50%
4	1	Domínio de 50% a 75%

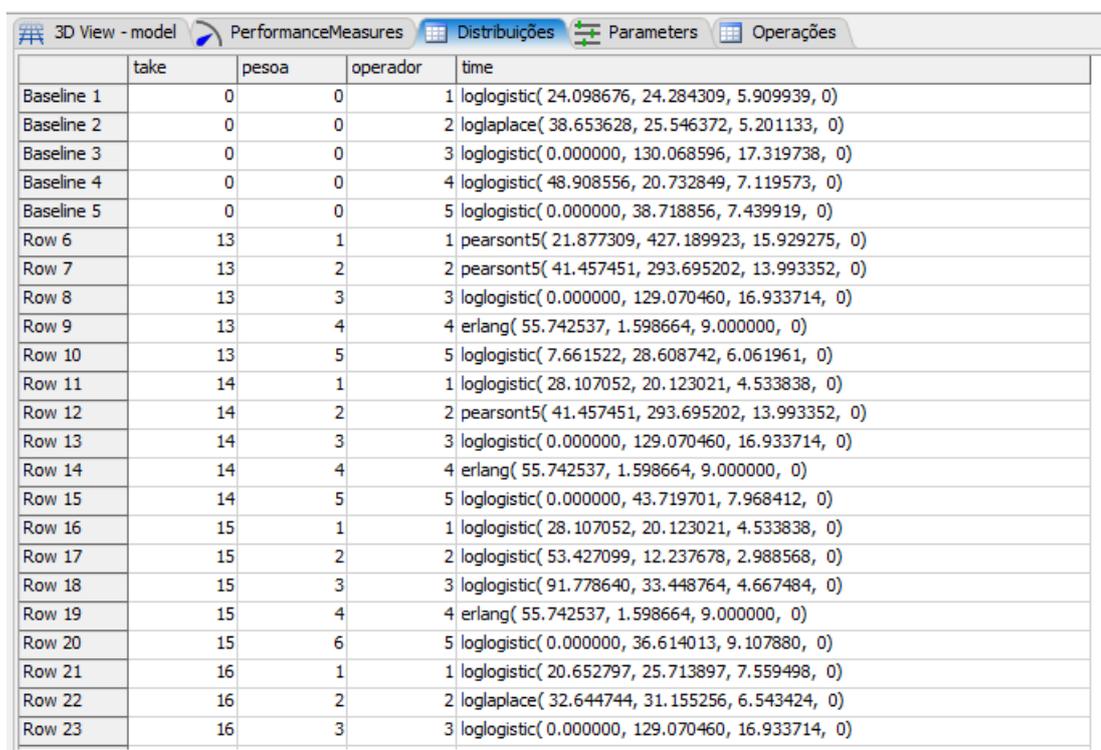
4	2	Domínio de 50% a 75%
4	3	Domínio de 50% a 75%
4	4	Domínio de 0% a 25%
4	5	Domínio de 50% a 75%
4	6	Domínio de 25% a 50%
4	7	Domínio de 25% a 50%
4	8	Domínio de 25% a 50%
4	9	Domínio de 75% a 100%
4	10	Domínio de 0% a 25%
4	11	Domínio de 25% a 50%
4	12	Domínio de 75% a 100%
5	0	Domínio de 25% a 50%
5	1	Antecipação de 25% a 50%
5	2	Antecipação de 75% a 100%
5	3	Antecipação de 0% a 25%
5	4	Antecipação de 50% a 75%
5	5	Antecipação de 75% a 100%
5	6	Antecipação de 50% a 75%
5	7	Antecipação de 25% a 50%
5	8	Antecipação de 50% a 75%
5	9	Antecipação de 25% a 50%
5	10	Antecipação de 75% a 100%
5	11	Antecipação de 25% a 50%
5	12	Antecipação de 75% a 100%
5	1	Desconexão de 75% a 100%
5	2	Desconexão de 25% a 50%
5	3	Desconexão de 0% a 25%
5	4	Desconexão de 25% a 50%
5	5	Desconexão de 0% a 25%
5	6	Desconexão de 25% a 50%
5	7	Desconexão de 25% a 50%
5	8	Desconexão de 50% a 75%
5	9	Desconexão de 50% a 75%
5	10	Desconexão de 25% a 50%
5	11	Desconexão de 50% a 75%
5	12	Desconexão de 0% a 25%

Fonte: Autor

4.4.2 Atualização do modelo de simulação

Para comparar os resultados nos dois cenários levantados, o modelo de simulação precisou ter seus tempos de operação atualizados. Para facilitar a simulação de cada cenário dentro do *software* Flexsim[®], foi criada uma *Global Table* para configurar as funções de distribuição de probabilidade de cada posto de trabalho. As primeiras três colunas foram preenchidas com valores de referência que não são utilizados pela simulação. Porém, na quarta coluna foram inseridas as respectivas funções do Flexsim[®] para gerar valores conforme as distribuições da Tabela 8 conforme o turno associado na Tabela 9. Para tal, foram inseridos em sequência as funções para os operadores de 1 a 5 no Cenário Base, seguidas pelas funções para o primeiro turno dos mesmos operadores no Cenário Proposto e subsequentemente para cada turno do Cenário Proposto. A ordem dos elementos nesta tabela é importante devido à maneira como os tempos foram programados na simulação. A Figura 31 apresenta como foram dispostas as primeiras linhas da tabela. O Anexo B apresenta a tabela completa de distribuições.

Figura 31 – *Global Table* de distribuições



	take	pesoa	operador	time
Baseline 1	0	0	1	logistic(24.098676, 24.284309, 5.909939, 0)
Baseline 2	0	0	2	loglaplace(38.653628, 25.546372, 5.201133, 0)
Baseline 3	0	0	3	loglogistic(0.000000, 130.068596, 17.319738, 0)
Baseline 4	0	0	4	loglogistic(48.908556, 20.732849, 7.119573, 0)
Baseline 5	0	0	5	loglogistic(0.000000, 38.718856, 7.439919, 0)
Row 6	13	1	1	pearson5(21.877309, 427.189923, 15.929275, 0)
Row 7	13	2	2	pearson5(41.457451, 293.695202, 13.993352, 0)
Row 8	13	3	3	loglogistic(0.000000, 129.070460, 16.933714, 0)
Row 9	13	4	4	erlang(55.742537, 1.598664, 9.000000, 0)
Row 10	13	5	5	loglogistic(7.661522, 28.608742, 6.061961, 0)
Row 11	14	1	1	loglogistic(28.107052, 20.123021, 4.533838, 0)
Row 12	14	2	2	pearson5(41.457451, 293.695202, 13.993352, 0)
Row 13	14	3	3	loglogistic(0.000000, 129.070460, 16.933714, 0)
Row 14	14	4	4	erlang(55.742537, 1.598664, 9.000000, 0)
Row 15	14	5	5	loglogistic(0.000000, 43.719701, 7.968412, 0)
Row 16	15	1	1	loglogistic(28.107052, 20.123021, 4.533838, 0)
Row 17	15	2	2	loglogistic(53.427099, 12.237678, 2.988568, 0)
Row 18	15	3	3	loglogistic(91.778640, 33.448764, 4.667484, 0)
Row 19	15	4	4	erlang(55.742537, 1.598664, 9.000000, 0)
Row 20	15	6	5	loglogistic(0.000000, 36.614013, 9.107880, 0)
Row 21	16	1	1	loglogistic(20.652797, 25.713897, 7.559498, 0)
Row 22	16	2	2	loglaplace(32.644744, 31.155256, 6.543424, 0)
Row 23	16	3	3	loglogistic(0.000000, 129.070460, 16.933714, 0)

Fonte: Autor.

No modelo de simulação, foi criado um parâmetro numérico chamado de "Turno". A esse parâmetro foi atribuído o valor zero para representar o Cenário Base e valores de 1 a 12

para representar os 12 turnos do Cenário Proposto. Da maneira como o parâmetro Turno e a *Global Table* de distribuições foram arranjados, a linha na tabela que representa um determinado operador em determinado turno pode ser obtida através da Equação 4.2, onde "linha" é o número da linha calculado, "turno" é o valor para o Parâmetro Turno da simulação e "operador" é o número do operador (de 1 a 5).

$$linha = turno \times 5 + operador \quad (4.2)$$

O cálculo dos tempos de operação dos postos de trabalho do operador 1 ao 5 foram ajustados para os valores abaixo respectivamente:

```
Table("Distribuicoes").executeCell(Model.parameters.Turno * 5 + 1, 4)
Table("Distribuicoes").executeCell(Model.parameters.Turno * 5 + 2, 4)
Table("Distribuicoes").executeCell(Model.parameters.Turno * 5 + 3, 4)
Table("Distribuicoes").executeCell(Model.parameters.Turno * 5 + 4, 4)
Table("Distribuicoes").executeCell(Model.parameters.Turno * 5 + 5, 4)
```

O modelo objeto deste estudo foi projetado para produzir 40 carrinhos em cada turno independentemente do tempo despendido para produzi-los. Para comparar os cenários estabelecidos optou-se por simular a primeira hora de cada turno. Assim, a quantidade produzida somando os 12 turnos pode ser comparada ao valor produzido na linha de montagem real. A próxima seção detalha como este modelo foi executado.

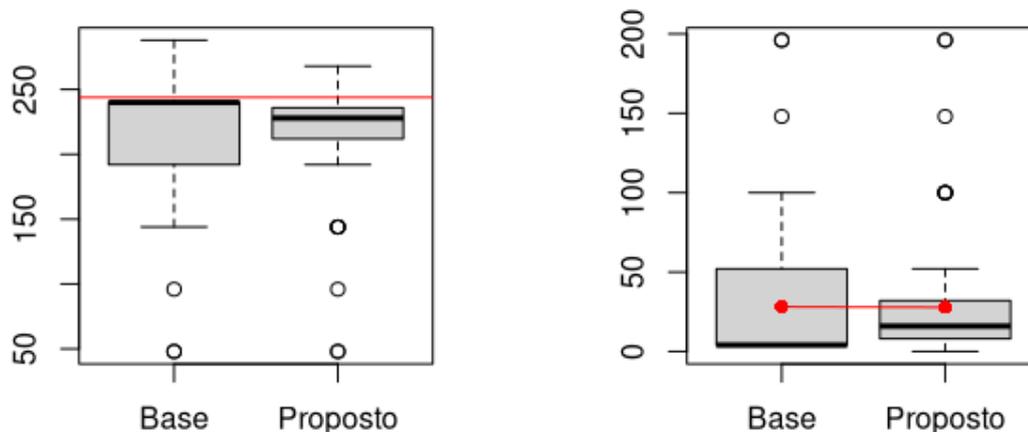
4.4.3 Execução do modelo e comparação dos resultados

O modelo foi executado em 100 iterações por cenário utilizando a ferramenta "Experimenter" do Flexsim[®]. A Tabela 10 apresenta os valores médios extraídos das 100 iterações em cada cenário e a Figura 32 apresenta a variação desses resultados.

Tabela 10 – Valos médios extraídos da simulação.

Cenário	Produção Média	Erro Médio
Base	220,32	28,08
Proposto	217,82	27,78

Fonte: Autor

Figura 32 – *Boxplots* dos resultados da simulação

(a) Quantidade total produzida

(b) Erro de estimativa

Fonte: Autor.

O valor total produzido na linha real foi aferido em 244 unidades e encontra-se representado pela linha vermelha na Figura 32a. A mesma figura apresenta a distribuição dos valores simulados nos cenários Base e Proposto. A diferença entre o valor produzido e o simulado em cada cenário compõe o erro da estimativa. Este erro é representado na Figura 32b. Nela também foi traçada a linha de tendência da média em vermelho.

Tanto na Tabela 10 quanto na Figura 32 observa-se uma pequena redução no erro de estimativa da simulação. Embora o valor médio da produção no cenário base seja mais próximo do valor real do que no cenário proposto, o erro médio no cenário proposto é menor. A razão para isso pode ser observada na Figura 32a. Embora os valores estimados no cenário base possuam uma variação maior em relação ao valor real, essa variação ocorre tanto para mais quanto para menos, equilibrando o valor médio mais próximo do valor real. Esse fenômeno ocorre com menos intensidade no cenário proposto. O valor médio do erro por outro lado é calculado a partir do valor absoluto da diferença entre o valor real e o estimado. Uma vez que os valores do cenário proposto são mais próximos ao valor real, o erro médio neste cenário é menor. Além disso, é possível notar uma dispersão menor dos resultados no cenário proposto tanto em relação à produção quanto em relação ao erro. Portanto, podemos afirmar que a hipótese 3 da seção 1.1 foi confirmada: um modelo de simulação cujas curvas de distribuição de probabilidade consideram

o humor do operador produz melhores resultados.

O próximo capítulo discorre sobre as implicações dos resultados obtidos neste capítulo.

5 CONSIDERAÇÕES FINAIS

O objetivo geral definido na seção 1.1 foi alcançado, uma vez que foram encontradas entre as variáveis extraídas com o modelo EMOTIC correlações significantes com o tempo de operação para todos os operadores da linha. Além disso, também foi demonstrada com sucesso a utilidade das variáveis em questão na modelagem de dados de entrada em um modelo de simulação.

Na seção 1.1 também foram levantadas três hipóteses para este projeto de pesquisa:

1. Um ou mais atributos de humor identificados pelo modelo de aprendizagem de máquina possuem correlação estatística com o tempo de operação.
2. A curva de distribuição de probabilidade dos tempos de processamento é diferente quando as amostras consideram classes diferentes de humor para um operador.
3. Um modelo de simulação cujas curvas de distribuição de probabilidade consideram o humor do operador produz melhores resultados.

A hipótese 1 foi confirmada na subseção 4.3.1. Nela, a Tabela 5 apresenta 24 variáveis cuja correlação com o tempo de operação apresentou no teste estatístico uma probabilidade acima de 95% de não ter ocorrido ao acaso. Algumas dessas variáveis apresentaram uma correlação tão significativa que o valor do *P-Value* precisou ser arredondado para zero devido a limitações computacionais. Portanto é possível afirmar com elevado grau de certeza que há correlação monotônica entre alguma dessas variáveis e o tempo de operação na linha *Karts Lego*[®]. Embora não seja possível generalizar esta afirmação a partir desta pesquisa, esta é uma contribuição significativa uma vez que evidencia a utilidade de modelos como o EMOTIC em mensurar fatores associados ao tempo de operação em linhas de montagem. Também foi possível identificar tanto na Tabela 5 quanto no Anexo A que a variável Domínio possui correlação com os tempos de operação de quase todos os operadores.

Na subseção 4.3.4, a Tabela 8 apresenta variações significativas nas curvas de distribuição de probabilidade para diferentes classes de humor. Essa variação é observada tanto em relação ao modelo de distribuição em alguns casos quanto em relação aos parâmetros do modelo em outro casos. Assim, valida-se também a hipótese 2.

Foram apresentadas na subseção 4.4.3 pequenas melhorias quanto à precisão do modelo de simulação objeto deste estudo. Embora este resultado não seja suficiente para validar a terceira hipótese de maneira generalizada, é uma evidência positiva para sua validação.

Quanto aos objetivos específicos, observando a validação das hipóteses acima, foi possível identificar as características de humor aparente através das filmagens da linha em operação. Múltiplas destas características, representadas por variáveis no modelo EMOTIC, apresentaram correlação significativa com o tempo de operação, tanto individualmente para cada operador, quanto para a linha como um todo.

A partir das variáveis selecionadas para o experimento, foi possível criar um modelo matemático cujo tempo de operação é afetado pelo humor aparente do operador e este modelo foi demonstrado no modelo de simulação desenvolvido por Rocha (2020). Por fim, os resultados do modelo antes e após as modificações foram comparados.

Cabe mencionar que os resultados e conclusões apresentados limitam-se à linha de produção e operadores objeto do estudo. Para que seja possível generalizar os resultados encontrados, é necessário um conjunto amostral maior. Este trabalho, no entanto, apresenta procedimentos que podem ser replicados em outras linhas de montagem para que um estudo mais abrangente possa ser realizado. Estes procedimentos, apresentados resumidamente na Figura 2 são uma das maiores contribuições deste trabalho uma vez que proporcionam um modelo base para que pesquisas similares possam ser conduzidas.

A confirmação de que existem variáveis correlatas ao tempo de operação (hipótese 1) é uma importante contribuição deste trabalho. Principalmente por haver evidências de que esta correlação independe do indivíduo e da tarefa sendo executada. Esta contribuição enriquece a acadêmica sobre fatores que podem vir a impactar o desempenho humano e abre caminho para mais trabalhos acadêmicos e aplicações práticas de modelos similares ao EMOTIC em diversas áreas incluindo simulações, saúde e segurança operacional.

Uma descoberta interessante envolvendo estas correlações é a natureza contraintuitiva de algumas. Por exemplo, a variável Dúvida / Confusão, se pensarmos intuitivamente, provavelmente iríamos sugerir uma correlação positiva com o tempo de operação, ou seja, quanto mais Dúvida / Confusão, maior o tempo do operador para finalizar sua tarefa. No entanto, o oposto ocorreu: foi encontrada uma correlação significativa negativa para esta variável no operador 2. De forma similar, Domínio apresentou uma correlação positiva para alguns operadores. Além disso,

variáveis que se esperaria produzir grande impacto no desempenho dos operadores, como a Fadiga, por exemplo, surpreendentemente, não apresentaram uma significância alta para a maioria dos operadores. Cabe salientar que embora as variáveis do modelo EMOTIC apresentaram estes comportamentos contraintuitivos, isto não quer dizer que estas características se comportem de fato desta forma. Isto pode ocorrer porque aquilo que o modelo identifica como Dúvida / Confusão, Domínio e Fadiga, por exemplo, pode não representar o que estas palavras realmente significam. Tais comportamentos também podem ocorrer devido a facilidade ou dificuldade do modelo em identificar determinadas variáveis a partir das filmagens.

A confirmação de que as curvas de distribuição de probabilidade diferem para diferentes classes de humor (hipótese 2), por sua vez, apresenta uma possível forma utilização da relação encontrada tais variáveis para fins mais práticos. Embora o método utilizado seja de certa forma rudimentar, é uma contribuição importante para a literatura acadêmica.

Ao aplicar tais curvas de distribuição de probabilidade em um modelo de simulação, a aplicabilidade deste trabalho foi demonstrada com ainda mais força. As melhorias nos resultados do modelo que confirmaram a hipótese 3, embora sejam pequenas, representam uma grande contribuição para a escassa literatura acadêmica no que envolve simulações e fatores ligados ao desempenho humano, principalmente quando se trata de um fator pessoal, como é o caso do humor.

A técnica proposta também preenche a lacuna na literatura apontada no Capítulo 3, uma vez que a aferição do humor não onera os operadores com carga adicional de trabalho nem exige equipamentos sofisticados ou pouco acessíveis.

Embora os procedimentos representados na Figura 2 sejam um bom *framework* inicial para o desenvolvimento de pesquisas similares a esta, alguns desses passos foram implementados de maneira muito rudimentar nesta pesquisa. Um exemplo é o passo 1.10 descrito na subseção 4.1.5. Este método baseado em um ponto central foi suficiente para o escopo desta pesquisa, porém depende da estimativa de um ponto central em um processo manual e iterativo moroso. Além disso, é necessário que o operador não se desloque significativamente dentro do posto de trabalho para que o método seja eficaz. Por fim, em casos em que os postos de trabalho são muito próximos uns dos outros, como ocorreu com o operador 2 nesta pesquisa, é esperado um nível elevado de erros de identificação.

Um outro passo que foi implementado de forma rudimentar é o 3.3 descrito na subse-

ção 4.3.3. As classes de humor definidas consideram faixas muito amplas de uma única variável contínua. Esta classificação é bastante simplificada e não tira proveito do amplo volume de informações obtido pelas etapas anteriores.

Por fim, esta pesquisa pode ser o passo inicial para um grande leque de pesquisas futuras. Os trabalhos futuros sugeridos são apresentados na seção a seguir.

5.1 Sugestões para trabalhos futuros

Nesta seção são apresentadas algumas sugestões para trabalhos futuros com base nos resultados desta pesquisa. Uma dessas melhorias diz respeito à subseção 4.3.2. Nela foi selecionada a variável com correlação mais significativa para cada operador, porém outras 62 variáveis com correlação estatisticamente significativa foram descartadas. É sugerido para trabalhos futuros o estudo dessas variáveis em relação ao tempo de operação.

Embora o método de identificação do operador baseado no ponto central apresentado na subseção 4.1.5 tenha se mostrado adequado para este trabalho, este é uma das principais fontes de imprecisão encontradas no decorrer do experimento. Além disso, como mencionado anteriormente, em outros contextos onde haja uma movimentação maior dos operadores este método pode não ser viável. A substituição do método utilizado por um que não possua tais limitações e a posterior comparação com os resultados obtidos pelo método original é sugerida como trabalho futuro. Um possível método substituto seria a utilização de um outro modelo de aprendizagem de máquina para identificar o indivíduo em cada retângulo onde um indivíduo foi detectado, identificando assim o operador.

A linha de produção objeto deste estudo é uma linha fictícia, criada para o ensino de cronoanálise. Embora a linha vise reproduzir cenários comumente encontrados em linhas reais, a aplicação dos procedimentos em linhas de montagem reais pode preencher lacunas não abordadas por esta pesquisa. Portanto esta aplicação é sugerida como trabalho futuro.

Como foi citado na seção anterior, os resultados desta pesquisa não são suficientes para tirar conclusões generalizadas. Portanto é sugerida a realização de pesquisas reproduzindo os procedimentos descritos nesta pesquisa em uma amostra maior de linhas de montagem, sejam elas reais ou fictícias, envolvendo um número maior de operadores distintos. Desta forma será possível tirar conclusões generalizadas quanto à aplicação do humor aparente em relação ao tempo de trabalho dos operadores.

Em trabalhos futuros também é possível estudar a correlação do humor aparente com outros aspectos da linha de produção. Estes aspectos incluem, mas não se limitam a incidência de defeitos, paradas não programadas, saúde ocupacional e segurança.

O modelo matemático utilizado neste trabalho para embutir o humor aparente no modelo de simulação cumpriu seu propósito demonstrativo, porém é bastante rudimentar. Em trabalhos futuros é sugerido um modelo mais elaborado que observe padrões de variação das variáveis de humor no tempo, por exemplo. Uma outra possibilidade é a observação de múltiplas variáveis ao mesmo tempo dentro do modelo. Isto é obtido mais facilmente em simulações baseadas em agentes ou híbridas.

Uma outra possível aplicação envolve a utilização de técnicas como cartas de controle para observar a variação das variáveis de humor aparente dos operadores. Tais pesquisas podem trazer benefícios envolvendo a saúde ocupacional dos operadores, prevenção de acidentes e de defeitos de fabricação.

Também é possível automatizar os procedimentos descritos nesta pesquisa de forma que sejam realizados em tempo real ou quase real. Desta forma, em trabalhos futuros, será possível observar as cartas de controle descritas anteriormente em tempo real, bem como a utilização destas variáveis em gêmeos digitais.

REFERÊNCIAS

- ADEWUNMI, A.; AICKELIN, U.; BANGSOW, S. **Use Cases of Discrete Event Simulation: Appliance and Research**. 1. ed. [S.l.]: Springer-Verlag Berlin Heidelberg, 2012. ISBN 978-3-642-28776-3,978-3-642-28777-0.
- AMARAL, J. V. S. D.; LIMA, R. D. S.; MIRANDA, R. D. C. Modelagem e simulação na cadeia de suprimentos do petróleo: revisão sistemática de literatura. In: **Anais do Encontro Nacional de Engenharia de Produção**. [S.l.]: ENEGEP 2020 - Encontro Nacional de Engenharia de Produção, 2020.
- AMARAL, J. V. S. do. **Otimização Baseada em Metamodelos: Uma Abordagem para Metamodelagem em Simulação a Eventos Discretos**. Dissertação (Mestrado) — Universidade Federal de Itajubá, 2021.
- AMARAL, J. V. S. do *et al.* Metamodel-based simulation optimization: A systematic literature review. **Simulation Modelling Practice and Theory**, Elsevier BV, v. 114, p. 102403, jan 2022.
- APPOLINÁRIO, F. **Metodologia da ciência: Filosofia e prática da pesquisa**. 2. ed. [S.l.]: Cengage Learning, 2012.
- BAINES, T. *et al.* Towards a theoretical framework for human performance modelling within manufacturing systems design. Elsevier BV, v. 13, n. 6, p. 486–504, sep 2005.
- BAINES, T. S. **Modelling in the evaluation of a manufacturing strategy**. Tese (Doutorado) — Cranfield University, 1994.
- BAINES, T. S.; KAY, J. M. Human performance modelling as an aid in the process of manufacturing system design: A pilot study. **International Journal of Production Research**, Informa UK Limited, v. 40, n. 10, p. 2321–2334, jan 2002. Cited By 50. Disponível em: <<https://www.scopus.com/inward/record.uri?eid=2-s2.0-0037054881&doi=10.1080%2f00207540210128198&partnerID=40&md5=e0cfdca8c7a2564cd692725f5add938a>>.
- BAKHTIAR, A. *et al.* Relationship of quality management system standards to industrial property rights in indonesia using spearman correlation analysis method. **IOP Conference Series: Earth and Environmental Science**, IOP Publishing, v. 623, n. 1, p. 012092, jan 2021.
- BANKS, J. **Discrete event system simulation**. [S.l.]: Pearson Education India, 2005.
- BERTRAND, J. W. M.; FRANSOO, J. C. Operations management research methodologies using quantitative modeling. **International Journal of Operations & Production Management**, Emerald, v. 22, n. 2, p. 241–264, feb 2002.
- BORGHETTI, B.; GIAMETTA, J.; RUSNOCK, C. Assessing continuous operator workload with a hybrid scaffolded neuroergonomic modeling approach. **Human Factors**, v. 59, n. 1, p. 134–146, 2017. Cited By 7. Disponível em: <<https://www.scopus.com/inward/record.uri?eid=2-s2.0-85011585267&doi=10.1177%2f0018720816672308&partnerID=40&md5=a68959d55f9e92692fedf3c0077e499b>>.
- BREIMAN, L. Random forests. **Machine Learning**, Springer Science and Business Media LLC, v. 45, n. 1, p. 5–32, 2001.

BUNTING, A.; BELYAVIN, A. Modelling human performance in semi-automated systems. In: **1999 International Conference on Human Interfaces in Control Rooms, Cockpits and Command Centres**. [S.l.]: IET, 1999.

CALLE, E. G. *et al.* Correlation analysis of sleep quality, mood and teleoperation performance in the MDRS206 analog mission. **Journal of Space Safety Engineering**, Elsevier BV, v. 8, n. 4, p. 312–316, dec 2021.

CHOI, S. Agent-based human-robot interaction simulation model for the analysis of operator performance in the supervisory control of ugvs. **International Journal of Precision Engineering and Manufacturing**, v. 19, n. 5, p. 685–693, 2018. Cited By 1. Disponível em: <<https://www.scopus.com/inward/record.uri?eid=2-s2.0-85047193608&doi=10.1007%2fs12541-018-0082-3&partnerID=40&md5=e81b571e1481a5c6d037cab10823115f>>.

DENYER, D.; TRANFIELD, D. Producing a systematic review. In: _____. **The Sage handbook of organizational research methods**. [S.l.]: SAGE Publications, Inc, 2008. p. 671–689.

DIGIESI, S. *et al.* The effect of dynamic worker behavior on flow line performance. **International Journal of Production Economics**, Elsevier BV, v. 120, n. 2, p. 368–377, aug 2009.

DU, C.-J.; SUN, D.-W. Learning techniques used in computer vision for food quality evaluation: a review. **Journal of Food Engineering**, Elsevier BV, v. 72, n. 1, p. 39–55, jan 2006.

ELSEVIER. **Scopus: Guia de Referência Rápida**. 2015. Disponível em: <https://www.periodicos.capes.gov.br/images/documents/Scopus_Guiadereferênciarápida_10.08.2016.pdf>.

GARDNER, M.; DORLING, S. Artificial neural networks (the multilayer perceptron)—a review of applications in the atmospheric sciences. **Atmospheric Environment**, Elsevier BV, v. 32, n. 14-15, p. 2627–2636, aug 1998.

GREASLEY, A. The case for the organisational use of simulation. **Journal of Manufacturing Technology Management**, Emerald, v. 15, n. 7, p. 560–566, oct 2004.

GREGORIADES, A. *et al.* Human-centered safety analysis of prospective road designs. **IEEE Transactions on Systems, Man, and Cybernetics Part A: Systems and Humans**, v. 40, n. 2, p. 236–250, 2010. Cited By 24. Disponível em: <<https://www.scopus.com/inward/record.uri?eid=2-s2.0-77249085111&doi=10.1109%2fTSMCA.2009.2037011&partnerID=40&md5=0a14e927898b642a4f0292d81e5fc51d>>.

HART, S. G.; STAVELAND, L. E. Development of NASA-TLX (task load index): Results of empirical and theoretical research. In: **Advances in Psychology**. [S.l.]: Elsevier, 1988. p. 139–183.

HEIMSTRA, N. W.; ELLINGSTAD, V. S.; DEKOCK, A. R. Effects of operator mood on performance in a simulated driving task. **Perceptual and Motor Skills**, SAGE Publications, v. 25, n. 3, p. 729–735, dec 1967.

HIRT, E. R. *et al.* Processing goals, task interest, and the mood–performance relationship: A mediational analysis. **Journal of Personality and Social Psychology**, American Psychological Association (APA), v. 71, n. 2, p. 245–261, 1996.

- HUGHES, K.; JIANG, X. Using discrete event simulation to model excavator operator performance. **Human Factors and Ergonomics in Manufacturing & Service Industries**, Wiley, v. 20, n. 5, p. 408–423, may 2010.
- JAIN, P. K.; PAMULA, R.; SRIVASTAVA, G. A systematic literature review on machine learning applications for consumer sentiment analysis using online reviews. **Computer Science Review**, Elsevier BV, v. 41, p. 100413, aug 2021.
- KAEHLING, L. P.; LITTMAN, M. L.; MOORE, A. W. Reinforcement learning: A survey. **Journal of Artificial Intelligence Research**, AI Access Foundation, v. 4, p. 237–285, may 1996.
- KOSTI, R. *et al.* Context based emotion recognition using EMOTIC dataset. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, Institute of Electrical and Electronics Engineers (IEEE), p. 1–1, 2019.
- KOSTI, R. *et al.* Emotic: Emotions in context dataset. In: **Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops**. [S.l.: s.n.], 2017.
- KUMAR, D. P.; AMGOTH, T.; ANNAVAPU, C. S. R. Machine learning algorithms for wireless sensor networks: A survey. Elsevier BV, v. 49, p. 1–25, sep 2019.
- LAW, A. M.; KELTON, W. D. **Simulation modeling and analysis**. [S.l.]: McGraw-Hill New York, 2000. v. 3.
- LEAL, F.; ALMEIDA, D. d.; MONTEVECHI, J. A. B. Uma proposta de técnica de modelagem conceitual para a simulação através de elementos do idef. **Simpósio Brasileiro de Pesquisa Operacional**, v. 40, p. 2503–2514, 2008.
- LEAL, F. *et al.* Elaboração de modelos conceituais em simulação computacional através de adaptações na técnica idef0: uma aplicação prática. **Atas do XXVII Encontro Nacional de Engenharia de Produção**, 2007.
- LECUN, Y.; BENGIO, Y.; HINTON, G. Deep learning. **Nature**, Springer Science and Business Media LLC, v. 521, n. 7553, p. 436–444, may 2015.
- LEVY, Y.; ELLIS, T. J. A systems approach to conduct an effective literature review in support of information systems research. **Informing Science**, v. 9, 2006.
- MAJDZIK, P. A feasible schedule for parallel assembly tasks in flexible manufacturing systems. **International Journal of Applied Mathematics and Computer Science**, v. 32, n. 1, p. 51–63, 2022. Cited By 0. Disponível em: <<https://www.scopus.com/inward/record.uri?eid=2-s2.0-85128209959&doi=10.34768%2famcs-2022-0005&partnerID=40&md5=934ac4cb61aade4805735507ef9d109d>>.
- MASON, S. *et al.* Improving the design process for factories: Modeling human performance variation. **Journal of Manufacturing Systems**, v. 24, n. 1, p. 47–54, 2005. Cited By 25. Disponível em: <<https://www.scopus.com/inward/record.uri?eid=2-s2.0-33744799755&doi=10.1016%2fs0278-6125%2805%2980006-8&partnerID=40&md5=9b6329d80b70da8131d5511b3a804913>>.
- MCCARTHY, J. What is artificial intelligence? 1998.

- MICHALSKI, R. S.; CARBONELL, J. G.; MITCHELL, T. M. **Machine Learning : an Artificial Intelligence Approach**. Berlin, Heidelberg: Springer Berlin Heidelberg, 1983. ISBN 9783662124055.
- MIGUEL, P. A. C. *et al.* **Metodologia de pesquisa em engenharia de produção e gestão de operações**. second. Rio de Janeiro, Brasil: Elsevier, 2012. ISBN 8535248919.
- MINER, A. G.; GLOMB, T. M. State mood, task performance, and behavior at work: A within-persons approach. **Organizational Behavior and Human Decision Processes**, Elsevier BV, v. 112, n. 1, p. 43–57, may 2010.
- MIRANDA, R. de C. *et al.* Increasing the efficiency in integer simulation optimization: Reducing the search space through data envelopment analysis and orthogonal arrays. **European Journal of Operational Research**, Elsevier BV, v. 262, n. 2, p. 673–681, oct 2017.
- MOHRI, M.; ROSTAMIZADEH, A.; TALWALKAR, A. **Foundations of machine learning**. Cambridge, MA: MIT Press, 2012. ISBN 026201825X.
- MONTEVECHI, J. A. B. *et al.* Conceptual modeling in simulation projects by mean adapted IDEF: An application in a brazilian tech company. In: **Proceedings of the 2010 Winter Simulation Conference**. [S.l.]: IEEE, 2010.
- MONTEVECHI, J. A. B. *et al.* Identification of the main methods used in simulation projects. In: **Proceedings of the 2015 Winter Simulation Conference**. [S.l.]: IEEE, 2015.
- MONTEVECHI, J. A. B. *et al.* Application of design of experiments on the simulation of a process in automotive industry. In: **2007 Winter Simulation Conference**. [S.l.]: IEEE, 2007.
- MONTGOMERY, D. **Introduction to linear regression analysis**. Hoboken, NJ: Wiley A John Wiley & Sons, Inc, 2012. ISBN 9780470542811.
- NEAL, R. M. **Bayesian Learning for Neural Networks**. 1. ed. [S.l.]: Springer-Verlag New York, 1996. (Lecture Notes in Statistics 118). ISBN 978-0-387-94724-2,978-1-4612-0745-0.
- NILSSON, N. **Principles of artificial intelligence**. Los Altos, Calif: Morgan Kaufmann Publishers, 1986. ISBN 0934613109.
- O’KANE, J.; SPENCELEY, J.; TAYLOR, R. Simulation as an essential tool for advanced manufacturing technology problems. **Journal of Materials Processing Technology**, Elsevier BV, v. 107, n. 1-3, p. 412–424, nov 2000.
- OLIVEIRA, J. B.; LIMA, R. S.; MONTEVECHI, J. A. B. Perspectives and relationships in supply chain simulation: A systematic literature review. **Simulation Modelling Practice and Theory**, Elsevier BV, v. 62, p. 166–191, mar 2016.
- OLIVEIRA, M. L. M. de. **Utilização da simulação híbrida para representar o fator humano em sistemas produtivos**. Tese (Doutorado) — Universidade Federal de Itajubá, 2017.
- PAES, V. de C. **Crawler de Faces na Web**. Dissertação (Mestrado) — Universidade Federal de Itajubá, 2012. Disponível em: <<https://repositorio.unifei.edu.br/jspui/handle/123456789/1099>>.

- PAIVA, C. *et al.* Human factors in an automotive discrete event simulation model [fator humano em um modelo de simulação a eventos discretos do setor automotivo]. **Acta Scientiarum - Technology**, v. 39, p. 615–622, 2017. Cited By 1. Disponível em: <<https://www.scopus.com/inward/record.uri?eid=2-s2.0-85038375761&doi=10.4025%2factascitechnol.v39i0.30446&partnerID=40&md5=bf5b0b4fbada7c0babf887e26aad0df0>>.
- PECEK, B.; KOVACIC, A. Business process management: Use of simulation in the public sector. **Economic Research-Ekonomska Istraživanja**, Informa UK Limited, v. 24, n. 1, p. 95–106, jan 2011.
- QIANG, W.; ZHONGLI, Z. Reinforcement learning model, algorithms and its application. In: **2011 International Conference on Mechatronic Science, Electric Engineering and Computer (MEC)**. [S.l.]: IEEE, 2011.
- QUINLAN, J. R. Induction of decision trees. **Machine Learning**, Springer Science and Business Media LLC, v. 1, n. 1, p. 81–106, mar 1986.
- REBEKIĆ, A. *et al.* Pearson's or Spearman's Correlation Coefficient - Which One to Use? **Poljoprivreda**, Faculty of Agrobiotechnical Sciences Osijek, v. 21, n. 2, p. 47–54, dec 2015.
- ROCHA, F. **Simulação e realidade virtual: uma pesquisa experimental em engenharia de produção**. Tese (Doutorado) — Universidade Federal de Itajubá, 2020. Disponível em: <<https://repositorio.unifei.edu.br/jspui/handle/123456789/2307>>.
- SANTOS, C. H. dos *et al.* Decision support in productive processes through DES and ABS in the digital twin era: a systematic literature review. **International Journal of Production Research**, Informa UK Limited, v. 60, n. 8, p. 2662–2681, mar 2021.
- SCHRIBER, T. J.; BRUNNER, D. T.; SMITH, J. S. Inside discrete-event simulation software: How it works and why it matters. In: **Proceedings of the Winter Simulation Conference 2014**. [S.l.]: IEEE, 2017.
- SHANMUGANATHAN, S. S. e. S. **Artificial Neural Network Modelling**. 1. ed. [S.l.]: Springer International Publishing, 2016. (Studies in Computational Intelligence 628). ISBN 978-3-319-28493-4, 978-3-319-28495-8.
- SUTTON, R. S.; BARTO, A. G. **Reinforcement Learning: An introduction**. [S.l.]: MIT Press Ltd, 2018. ISBN 0262039249.
- TAYLOR, S. J. E.; ROBINSON, S. So where to next? a survey of the future for discrete-event simulation. Informa UK Limited, v. 1, n. 1, p. 1–6, dec 2006.
- TFHUB. **Object detection model trained on Open Images V4 with ImageNet pre-trained Inception Resnet V2 as image feature extractor**. 2021. Acessado: 23 jul. 2021. Disponível em: <https://tfhub.dev/google/faster_rcnn/openimages_v4/inception_resnet_v2/1>. Acesso em: 23 jul. 2021.
- TORRES, A. F. **A influência de diferentes estratégias de cronometragem e modelagem detempos na validação operacional de um modelo de simulação**. Dissertação (Mestrado) — UNIVERSIDADE FEDERAL DE ITAJUBÁ, 2017.

VILELA, F. *et al.* Effect of human factor performance on the productivity of a manual assembly line. **International Journal of Simulation Modelling**, v. 19, n. 3, p. 365–374, 2020. Cited By 1. Disponível em: <<https://www.scopus.com/inward/record.uri?eid=2-s2.0-85095809008&doi=10.2507%2fIJSIMM19-3-508&partnerID=40&md5=dcc41f25dace6d36d30017191a130e59>>.

VILELA, F. F. **Modelagem do ritmo de trabalho humano em um projeto de simulação através da criação de cenários com múltiplas distribuições**. Dissertação (Mestrado) — Universidade Federal de Itajubá, 2015.

XIAO, Y.; WATSON, M. Guidance on conducting a systematic literature review. **Journal of Planning Education and Research**, SAGE Publications, v. 39, n. 1, p. 93–112, aug 2017.

ZHANG, J.-H. *et al.* Modeling and control of operator functional state in a unified framework of fuzzy inference petri nets. **Computer Methods and Programs in Biomedicine**, v. 144, p. 147–163, 2017. Cited By 15. Disponível em: <<https://www.scopus.com/inward/record.uri?eid=2-s2.0-85017023741&doi=10.1016%2fj.cmpb.2017.03.016&partnerID=40&md5=61e16378e02860922498cf0401d8936a>>.

Apêndices

APÊNDICE A – CORRELAÇÕES

Tabela 11 – Correlação da emoção aparente com a duração do processo.

Operador	Tipo da Variável	Variável	Correlação	<i>P-Value</i>	Ocorrências
Todos	Discreta	Desconexão	-0,40005	0	1.499.025
Todos	Discreta	Felicidade	0,29942	0	10.886
Todos	Discreta	Antecipação	0,29253	0	148.144
Todos	Discreta	Entusiasmo	0,26841	0	2.434
Todos	Discreta	Incômodo	0,22367	0	45.261
Todos	Contínua	Excitação	0,2212	0	1.778.570
Todos	Discreta	Estima	0,21885	0	904
Todos	Discreta	Surpresa	0,20096	0	33.816
Todos	Discreta	Vergonha	0,20062	0	1.893
Todos	Discreta	Confiança	0,17319	0	41.745
Todos	Discreta	Dúvida / Confusão	-0,17265	0	1.513.423
Todos	Discreta	Sofrimento	0,15859	$7,994 \times 10^{-15}$	9.194
Todos	Discreta	Desaprovação	0,157	$1,488 \times 10^{-14}$	55.318
Todos	Contínua	Domínio	0,14313	$2,523 \times 10^{-12}$	1.778.570
Todos	Contínua	Valência	0,12088	$3,534 \times 10^{-09}$	1.778.570
Todos	Discreta	Raiva	0,11676	$1,185 \times 10^{-08}$	439
Todos	Discreta	Inquietação	-0,10535	$2,727 \times 10^{-07}$	913.683
Todos	Discreta	Paz	0,09421	$4,321 \times 10^{-06}$	4.103
Todos	Discreta	Engajamento	-0,07525	$2,449 \times 10^{-04}$	1.776.204
Todos	Discreta	Aversão	0,07437	$2,896 \times 10^{-04}$	178
Todos	Discreta	Fadiga	0,07332	$3,526 \times 10^{-04}$	250.689
Todos	Discreta	Prazer	0,07015	$6,308 \times 10^{-04}$	1.107
Todos	Discreta	Anseio	0,03973	$5,306 \times 10^{-02}$	52
Todos	Discreta	Empatia	0,03748	$6,804 \times 10^{-02}$	26
Todos	Discreta	Dor	0,03092	$1,323 \times 10^{-01}$	731
Todos	Discreta	Afeto	-0,0246	$2,312 \times 10^{-01}$	8
Todos	Discreta	Tristeza	-0,00415	$8,398 \times 10^{-01}$	67.005
Todos	Discreta	Medo	-0,00323	$8,751 \times 10^{-01}$	252
Todos	Discreta	Sensibilidade	-	-	0
1	Discreta	Antecipação	-0,19524	$1,344 \times 10^{-05}$	21.305
1	Contínua	Excitação	0,16627	$2,182 \times 10^{-04}$	245.327
1	Contínua	Domínio	-0,16554	$2,328 \times 10^{-04}$	245.327
1	Discreta	Desconexão	0,14171	$1,662 \times 10^{-03}$	220.407

1	Discreta	Prazer	0,10927	$1,552 \times 10^{-02}$	77
1	Discreta	Fadiga	-0,1021	$2,381 \times 10^{-02}$	48.178
1	Discreta	Incômodo	0,10086	$2,558 \times 10^{-02}$	6.480
1	Contínua	Valência	-0,09547	$3,463 \times 10^{-02}$	245.327
1	Discreta	Inquietação	0,08701	$5,426 \times 10^{-02}$	116.928
1	Discreta	Tristeza	-0,08226	$6,888 \times 10^{-02}$	13.311
1	Discreta	Surpresa	0,06813	$1,321 \times 10^{-01}$	4.929
1	Discreta	Paz	0,06734	$1,366 \times 10^{-01}$	24
1	Discreta	Raiva	0,05552	$2,199 \times 10^{-01}$	145
1	Discreta	Felicidade	-0,05282	$2,432 \times 10^{-01}$	1.013
1	Discreta	Entusiasmo	-0,04978	$2,714 \times 10^{-01}$	165
1	Discreta	Vergonha	0,04839	$2,851 \times 10^{-01}$	402
1	Discreta	Desaprovação	0,04467	$3,237 \times 10^{-01}$	9.201
1	Discreta	Confiança	0,0437	$3,343 \times 10^{-01}$	2.285
1	Discreta	Sofrimento	0,04365	$3,349 \times 10^{-01}$	2.560
1	Discreta	Dúvida / Confusão	0,03649	$4,203 \times 10^{-01}$	203.518
1	Discreta	Aversão	0,03374	$4,561 \times 10^{-01}$	41
1	Discreta	Medo	0,03158	$4,855 \times 10^{-01}$	96
1	Discreta	Dor	0,02749	$5,439 \times 10^{-01}$	344
1	Discreta	Engajamento	-0,02227	$6,229 \times 10^{-01}$	244.406
1	Discreta	Estima	0,01085	$8,106 \times 10^{-01}$	36
1	Discreta	Empatia	-0,00986	$8,276 \times 10^{-01}$	3
1	Discreta	Afeto	-0,00846	$8,518 \times 10^{-01}$	4
1	Discreta	Sensibilidade	-	-	0
1	Discreta	Anseio	-	-	0
2	Discreta	Dúvida / Confusão	-0,13306	$3,931 \times 10^{-03}$	352.953
2	Contínua	Valência	0,09929	$3,175 \times 10^{-02}$	386.469
2	Discreta	Vergonha	0,09537	$3,917 \times 10^{-02}$	694
2	Discreta	Empatia	0,09446	$4,109 \times 10^{-02}$	18
2	Discreta	Felicidade	0,09182	$4,712 \times 10^{-02}$	4.698
2	Discreta	Engajamento	-0,08879	$5,493 \times 10^{-02}$	385.628
2	Discreta	Estima	0,08034	$8,254 \times 10^{-02}$	164
2	Discreta	Inquietação	-0,07167	$1,215 \times 10^{-01}$	242.389
2	Discreta	Entusiasmo	0,0691	$1,355 \times 10^{-01}$	153
2	Discreta	Prazer	0,06894	$1,365 \times 10^{-01}$	345
2	Discreta	Dor	0,06298	$1,738 \times 10^{-01}$	301
2	Contínua	Domínio	0,06025	$1,933 \times 10^{-01}$	386.469
2	Discreta	Confiança	0,05887	$2,036 \times 10^{-01}$	1.735
2	Discreta	Antecipação	0,05658	$2,218 \times 10^{-01}$	23.093

2	Discreta	Sufrimento	0,05603	$2,263 \times 10^{-01}$	4.195
2	Discreta	Fadiga	-0,05517	$2,335 \times 10^{-01}$	85.774
2	Discreta	Paz	0,05239	$2,580 \times 10^{-01}$	55
2	Discreta	Incômodo	0,05019	$2,786 \times 10^{-01}$	15.371
2	Discreta	Aversão	-0,03717	$4,225 \times 10^{-01}$	28
2	Discreta	Tristeza	0,02581	$5,775 \times 10^{-01}$	23.625
2	Contínua	Excitação	-0,02129	$6,459 \times 10^{-01}$	386.469
2	Discreta	Raiva	0,0102	$8,258 \times 10^{-01}$	107
2	Discreta	Medo	0,00856	$8,535 \times 10^{-01}$	133
2	Discreta	Surpresa	0,00256	$9,559 \times 10^{-01}$	21.634
2	Discreta	Desaprovação	-0,00207	$9,643 \times 10^{-01}$	22.820
2	Discreta	Desconexão	0,00014	$9,976 \times 10^{-01}$	331.309
2	Discreta	Afeto	-	-	3
2	Discreta	Sensibilidade	-	-	0
2	Discreta	Anseio	-	-	1
3	Contínua	Domínio	0,25696	$1,291 \times 10^{-08}$	372.282
3	Discreta	Inquietação	-0,20482	$6,649 \times 10^{-06}$	216.434
3	Contínua	Valência	0,1745	$1,300 \times 10^{-04}$	372.282
3	Discreta	Incômodo	-0,16602	$2,749 \times 10^{-04}$	12.237
3	Discreta	Dúvida / Confusão	-0,16165	$3,990 \times 10^{-04}$	341.154
3	Discreta	Estima	0,14235	$1,850 \times 10^{-03}$	236
3	Discreta	Sufrimento	-0,1363	$2,885 \times 10^{-03}$	937
3	Discreta	Confiança	0,13604	$2,939 \times 10^{-03}$	4.162
3	Discreta	Desaprovação	-0,1358	$2,990 \times 10^{-03}$	10.835
3	Discreta	Antecipação	0,1323	$3,833 \times 10^{-03}$	42.885
3	Discreta	Tristeza	-0,12999	$4,501 \times 10^{-03}$	8.057
3	Discreta	Surpresa	-0,11798	$9,988 \times 10^{-03}$	2.966
3	Discreta	Desconexão	-0,10202	$2,603 \times 10^{-02}$	294.922
3	Discreta	Empatia	0,09567	$3,693 \times 10^{-02}$	4
3	Discreta	Felicidade	0,08303	$7,033 \times 10^{-02}$	3.848
3	Contínua	Excitação	0,05626	$2,205 \times 10^{-01}$	372.282
3	Discreta	Dor	-0,05036	$2,729 \times 10^{-01}$	36
3	Discreta	Vergonha	0,03431	$4,551 \times 10^{-01}$	735
3	Discreta	Prazer	-0,02473	$5,904 \times 10^{-01}$	609
3	Discreta	Aversão	0,02291	$6,180 \times 10^{-01}$	37
3	Discreta	Paz	-0,02271	$6,212 \times 10^{-01}$	77
3	Discreta	Medo	0,02257	$6,232 \times 10^{-01}$	16
3	Discreta	Entusiasmo	0,01422	$7,570 \times 10^{-01}$	968
3	Discreta	Raiva	-0,01157	$8,013 \times 10^{-01}$	98

3	Discreta	Engajamento	-0,00162	$9,719 \times 10^{-01}$	372.012
3	Discreta	Fadiga	-0,00128	$9,777 \times 10^{-01}$	59.075
3	Discreta	Afeto	-	-	1
3	Discreta	Sensibilidade	-	-	0
3	Discreta	Anseio	-	-	0
4	Contínua	Domínio	0,16881	$2,293 \times 10^{-04}$	385.903
4	Discreta	Paz	0,12137	$8,300 \times 10^{-03}$	3.762
4	Discreta	Desconexão	-0,11489	$1,250 \times 10^{-02}$	296.266
4	Contínua	Valência	0,11284	$1,417 \times 10^{-02}$	385.903
4	Discreta	Inquietação	-0,10528	$2,217 \times 10^{-02}$	125.174
4	Discreta	Dúvida / Confusão	-0,09878	$3,190 \times 10^{-02}$	265.746
4	Discreta	Tristeza	-0,08864	$5,431 \times 10^{-02}$	12.638
4	Discreta	Desaprovação	-0,06821	$1,389 \times 10^{-01}$	6.423
4	Contínua	Excitação	0,06549	$1,555 \times 10^{-01}$	385.903
4	Discreta	Fadiga	-0,06537	$1,562 \times 10^{-01}$	26.729
4	Discreta	Antecipação	0,06431	$1,630 \times 10^{-01}$	38.806
4	Discreta	Anseio	0,05369	$2,444 \times 10^{-01}$	42
4	Discreta	Vergonha	-0,05008	$2,775 \times 10^{-01}$	13
4	Discreta	Incômodo	-0,04325	$3,484 \times 10^{-01}$	4.062
4	Discreta	Confiança	0,03084	$5,039 \times 10^{-01}$	22.190
4	Discreta	Surpresa	0,02601	$5,730 \times 10^{-01}$	3.238
4	Discreta	Entusiasmo	-0,02017	$6,621 \times 10^{-01}$	939
4	Discreta	Sofrimento	0,0166	$7,191 \times 10^{-01}$	761
4	Discreta	Felicidade	-0,01543	$7,382 \times 10^{-01}$	756
4	Discreta	Estima	-0,01252	$7,862 \times 10^{-01}$	330
4	Discreta	Engajamento	-0,01177	$7,986 \times 10^{-01}$	385.810
4	Discreta	Prazer	-0,00652	$8,876 \times 10^{-01}$	47
4	Discreta	Afeto	-	-	0
4	Discreta	Raiva	-	-	4
4	Discreta	Aversão	-	-	11
4	Discreta	Medo	-	-	4
4	Discreta	Dor	-	-	15
4	Discreta	Sensibilidade	-	-	0
4	Discreta	Empatia	-	-	1
5	Discreta	Antecipação	0,33231	$1,883 \times 10^{-13}$	22.055
5	Discreta	Desconexão	-0,30163	$3,091 \times 10^{-11}$	356.121
5	Contínua	Domínio	0,28036	$7,589 \times 10^{-10}$	388.589
5	Contínua	Valência	0,25698	$1,895 \times 10^{-08}$	388.589
5	Discreta	Confiança	0,25667	$1,974 \times 10^{-08}$	11.373

5	Discreta	Inquietação	-0,21048	$4,696 \times 10^{-06}$	212.758
5	Discreta	Dúvida / Confusão	-0,17123	$2,073 \times 10^{-04}$	350.052
5	Discreta	Estima	0,15364	$8,876 \times 10^{-04}$	138
5	Discreta	Fadiga	0,12878	$5,416 \times 10^{-03}$	30.933
5	Discreta	Surpresa	0,12152	$8,715 \times 10^{-03}$	1.049
5	Discreta	Felicidade	0,10722	$2,074 \times 10^{-02}$	571
5	Discreta	Sufrimento	0,09348	$4,393 \times 10^{-02}$	741
5	Discreta	Dor	0,0926	$4,596 \times 10^{-02}$	35
5	Discreta	Paz	-0,08404	$7,021 \times 10^{-02}$	185
5	Contínua	Excitação	0,07963	$8,629 \times 10^{-02}$	388.589
5	Discreta	Incômodo	-0,07862	$9,039 \times 10^{-02}$	7.111
5	Discreta	Tristeza	0,07839	$9,134 \times 10^{-02}$	9.374
5	Discreta	Prazer	0,06809	$1,427 \times 10^{-01}$	29
5	Discreta	Vergonha	0,03919	$3,992 \times 10^{-01}$	49
5	Discreta	Desaprovação	-0,03172	$4,950 \times 10^{-01}$	6.039
5	Discreta	Aversão	-0,02963	$5,239 \times 10^{-01}$	61
5	Discreta	Engajamento	-0,02802	$5,467 \times 10^{-01}$	388.348
5	Discreta	Raiva	0,02514	$5,887 \times 10^{-01}$	85
5	Discreta	Entusiasmo	0,01889	$6,845 \times 10^{-01}$	209
5	Discreta	Afeto	-	-	0
5	Discreta	Medo	-	-	3
5	Discreta	Sensibilidade	-	-	0
5	Discreta	Empatia	-	-	0
5	Discreta	Anseio	-	-	9

Fonte: Autor

APÊNDICE B – FUNÇÕES PARA O TEMPO DE OPERAÇÃO

Tabela 12 – Funções para o tempo de operação no Flexsim

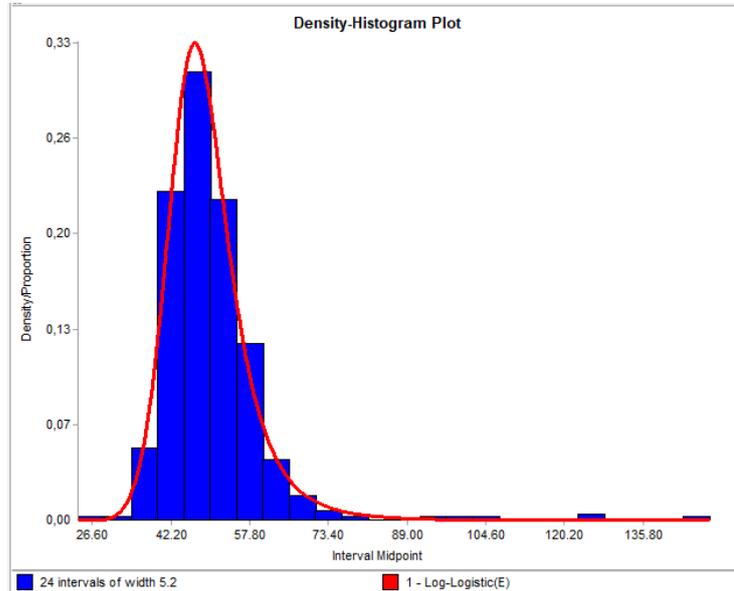
Turno	Operador	Função para o tempo de operação
0	1	loglogistic(24.098676, 24.284309, 5.909939, 0)
0	2	loglaplace(38.653628, 25.546372, 5.201133, 0)
0	3	loglogistic(0.000000, 130.068596, 17.319738, 0)
0	4	loglogistic(48.908556, 20.732849, 7.119573, 0)
0	5	loglogistic(0.000000, 38.718856, 7.439919, 0)
1	1	pearsont5(21.877309, 427.189923, 15.929275, 0)
1	2	pearsont5(41.457451, 293.695202, 13.993352, 0)
1	3	loglogistic(0.000000, 129.070460, 16.933714, 0)
1	4	erlang(55.742537, 1.598664, 9.000000, 0)
1	5	loglogistic(7.661522, 28.608742, 6.061961, 0)
2	1	loglogistic(28.107052, 20.123021, 4.533838, 0)
2	2	pearsont5(41.457451, 293.695202, 13.993352, 0)
2	3	loglogistic(0.000000, 129.070460, 16.933714, 0)
2	4	erlang(55.742537, 1.598664, 9.000000, 0)
2	5	loglogistic(0.000000, 43.719701, 7.968412, 0)
3	1	loglogistic(28.107052, 20.123021, 4.533838, 0)
3	2	loglogistic(53.427099, 12.237678, 2.988568, 0)
3	3	loglogistic(91.778640, 33.448764, 4.667484, 0)
3	4	erlang(55.742537, 1.598664, 9.000000, 0)
3	5	loglogistic(0.000000, 36.614013, 9.107880, 0)
4	1	loglogistic(20.652797, 25.713897, 7.559498, 0)
4	2	loglaplace(32.644744, 31.155256, 6.543424, 0)
4	3	loglogistic(0.000000, 129.070460, 16.933714, 0)
4	4	loglogistic(43.242559, 24.950912, 8.112738, 0)
4	5	pearsont6(0.000000, 25.693908, 40.327079, 26.660229, 0)
5	1	loglogistic(28.107052, 20.123021, 4.533838, 0)
5	2	loglaplace(32.644744, 31.155256, 6.543424, 0)
5	3	loglogistic(0.000000, 131.219083, 18.238980, 0)
5	4	erlang(55.742537, 1.598664, 9.000000, 0)
5	5	loglogistic(0.000000, 43.719701, 7.968412, 0)
6	1	loglogistic(28.828372, 20.743285, 4.994389, 0)
6	2	pearsont5(41.457451, 293.695202, 13.993352, 0)

6	3	loglogistic(0.000000, 131.219083, 18.238980, 0)
6	4	loglogistic(52.238377, 17.876636, 5.969747, 0)
6	5	pearsont6(0.000000, 25.693908, 40.327079, 26.660229, 0)
7	1	loglogistic(28.107052, 20.123021, 4.533838, 0)
7	2	loglaplace(32.644744, 31.155256, 6.543424, 0)
7	3	loglogistic(0.000000, 129.070460, 16.933714, 0)
7	4	loglogistic(52.238377, 17.876636, 5.969747, 0)
7	5	loglogistic(7.661522, 28.608742, 6.061961, 0)
8	1	loglogistic(20.652797, 25.713897, 7.559498, 0)
8	2	pearsont5(41.457451, 293.695202, 13.993352, 0)
8	3	loglogistic(0.000000, 131.219083, 18.238980, 0)
8	4	loglogistic(52.238377, 17.876636, 5.969747, 0)
8	5	pearsont6(0.000000, 25.693908, 40.327079, 26.660229, 0)
9	1	loglogistic(28.107052, 20.123021, 4.533838, 0)
9	2	loglogistic(53.427099, 12.237678, 2.988568, 0)
9	3	loglogistic(87.043552, 46.654642, 7.154211, 0)
9	4	pearsont5(52.856056, 235.134356, 13.653457, 0)
9	5	loglogistic(7.661522, 28.608742, 6.061961, 0)
10	1	loglogistic(28.107052, 20.123021, 4.533838, 0)
10	2	loglaplace(32.644744, 31.155256, 6.543424, 0)
10	3	loglogistic(0.000000, 131.219083, 18.238980, 0)
10	4	loglogistic(43.242559, 24.950912, 8.112738, 0)
10	5	loglogistic(0.000000, 43.719701, 7.968412, 0)
11	1	loglogistic(28.107052, 20.123021, 4.533838, 0)
11	2	loglogistic(53.427099, 12.237678, 2.988568, 0)
11	3	loglogistic(0.000000, 131.219083, 18.238980, 0)
11	4	loglogistic(52.238377, 17.876636, 5.969747, 0)
11	5	loglogistic(7.661522, 28.608742, 6.061961, 0)
12	1	loglogistic(20.652797, 25.713897, 7.559498, 0)
12	2	loglaplace(32.644744, 31.155256, 6.543424, 0)
12	3	loglogistic(0.000000, 129.070460, 16.933714, 0)
12	4	pearsont5(52.856056, 235.134356, 13.653457, 0)
12	5	loglogistic(0.000000, 43.719701, 7.968412, 0)

Fonte: Autor

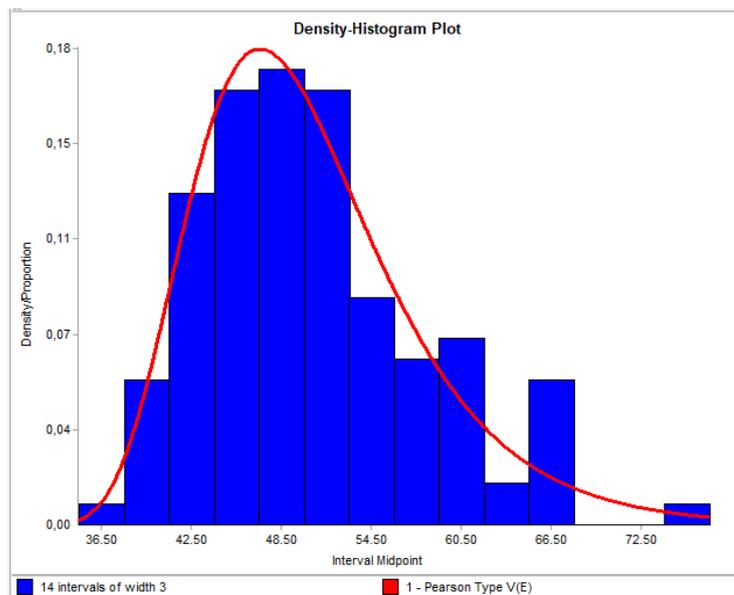
APÊNDICE C – HISTOGRAMAS

Figura 33 – Histograma da classe base para o operador 1



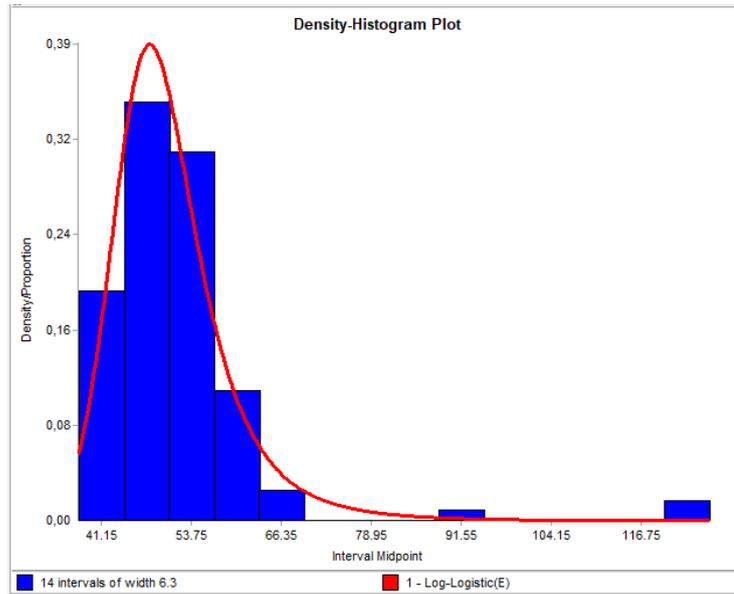
Fonte: Autor.

Figura 34 – Histograma da classe para o operador 1 de 0% a 25%



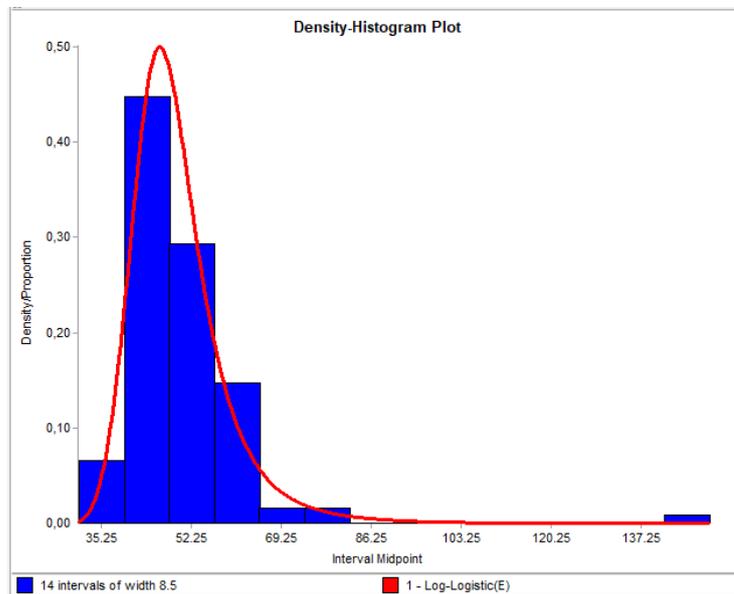
Fonte: Autor.

Figura 35 – Histograma da classe para o operador 1 de 25% a 50%



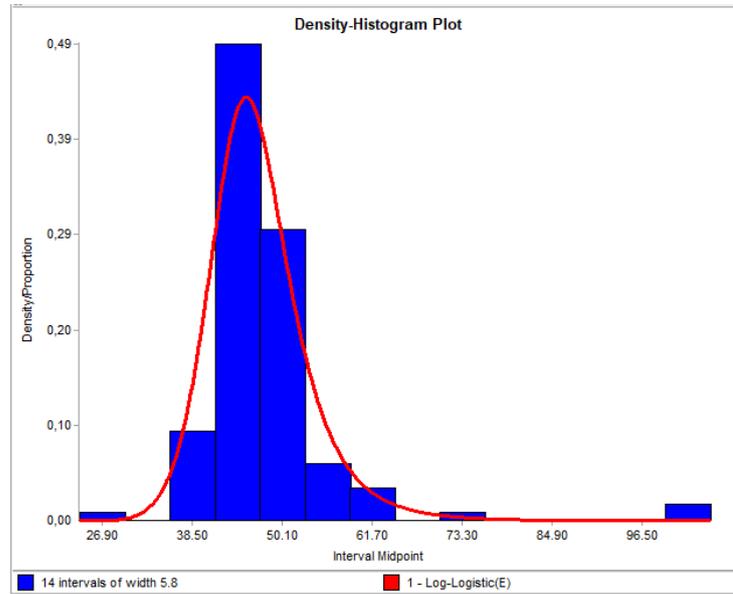
Fonte: Autor.

Figura 36 – Histograma da classe para o operador 1 de 50% a 75%



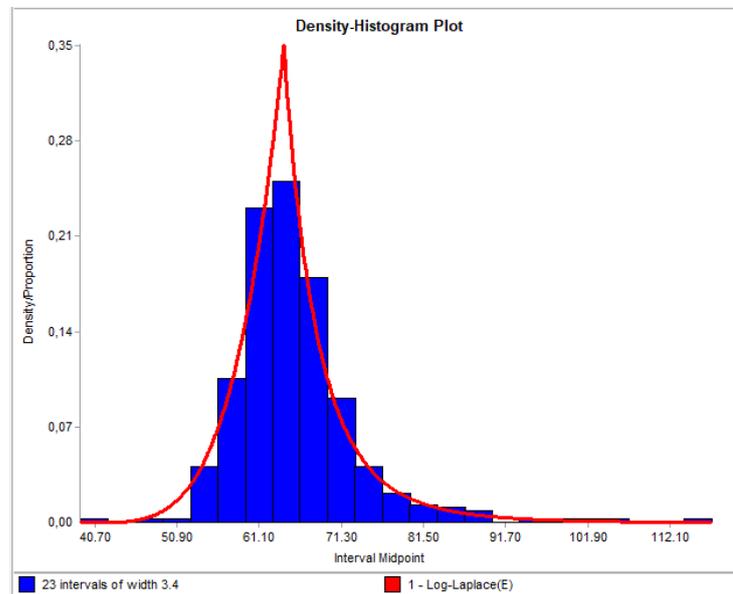
Fonte: Autor.

Figura 37 – Histograma da classe para o operador 1 de 75% a 100%



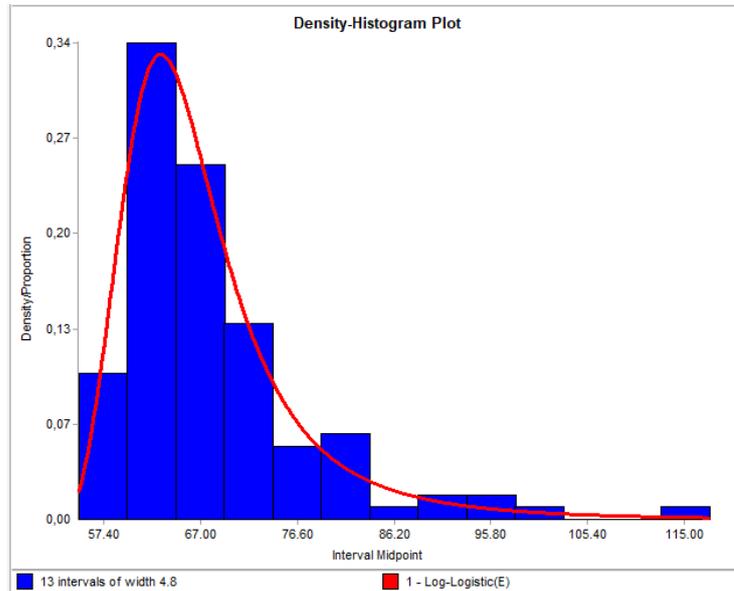
Fonte: Autor.

Figura 38 – Histograma da classe base para o operador 2



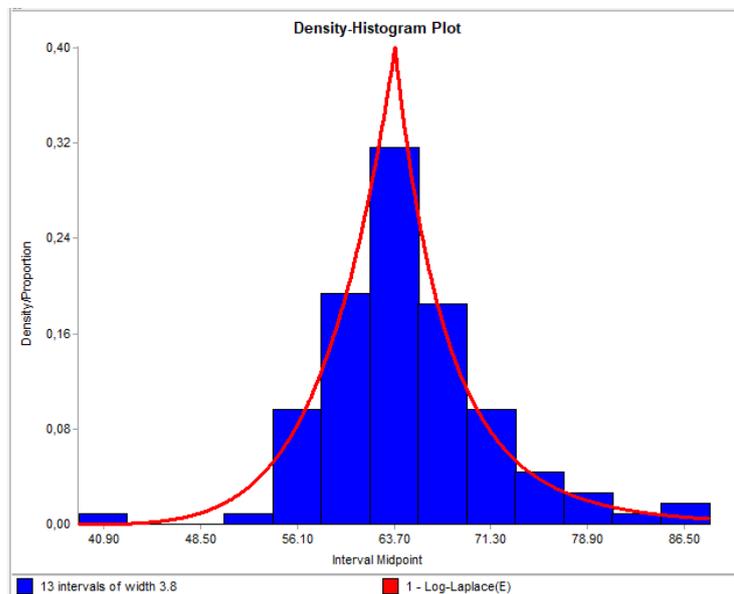
Fonte: Autor.

Figura 39 – Histograma da classe para o operador 2 de 0% a 25%



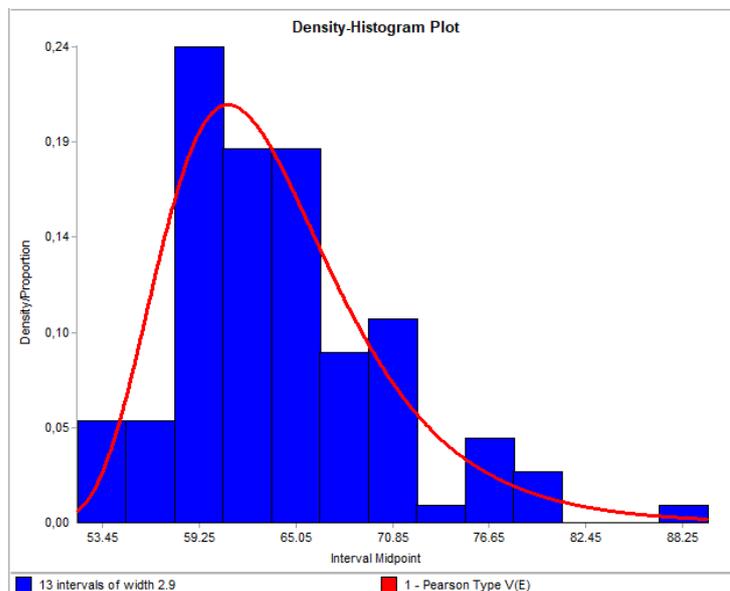
Fonte: Autor.

Figura 40 – Histograma da classe para o operador 2 de 25% a 50%



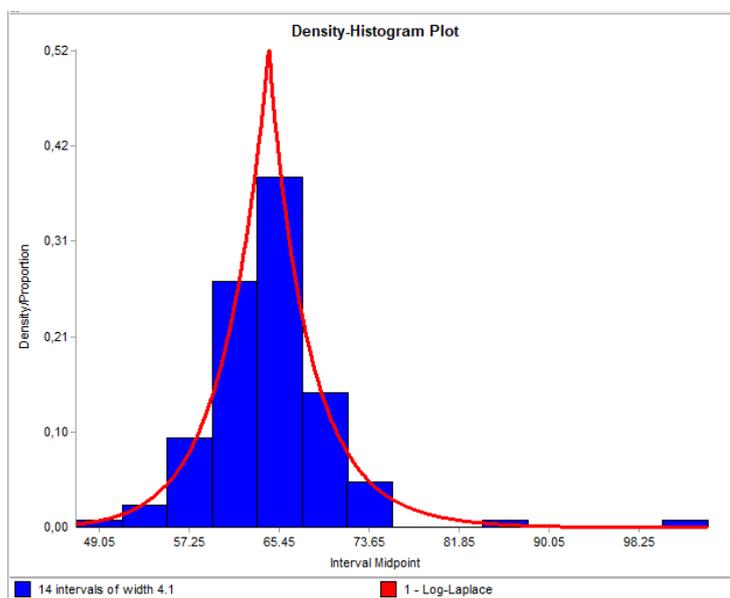
Fonte: Autor.

Figura 41 – Histograma da classe para o operador 2 de 50% a 75%



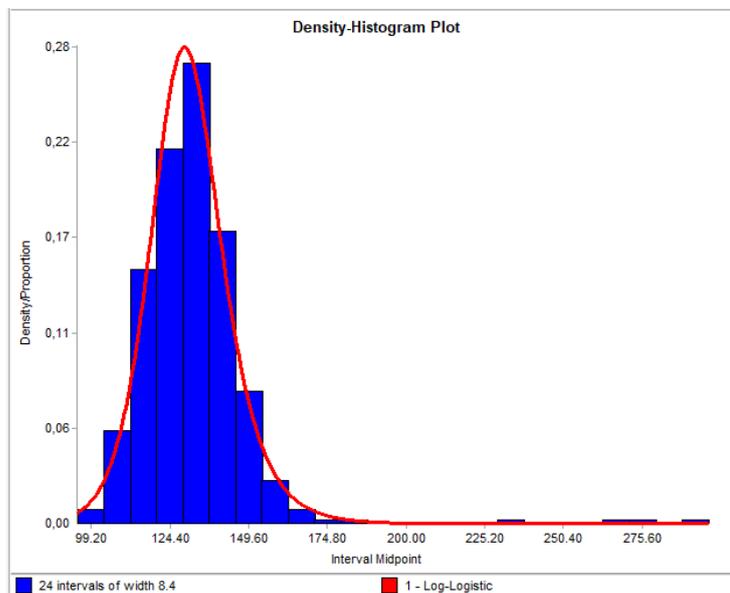
Fonte: Autor.

Figura 42 – Histograma da classe para o operador 2 de 75% a 100%



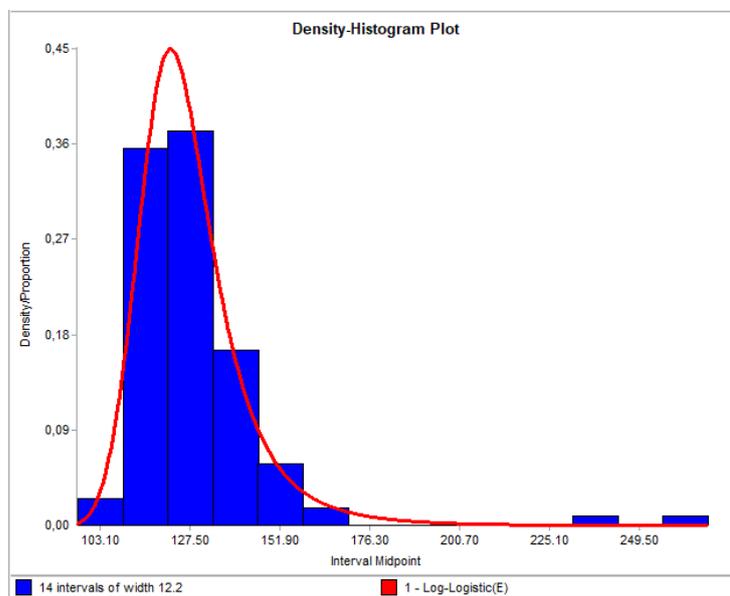
Fonte: Autor.

Figura 43 – Histograma da classe base para o operador 3



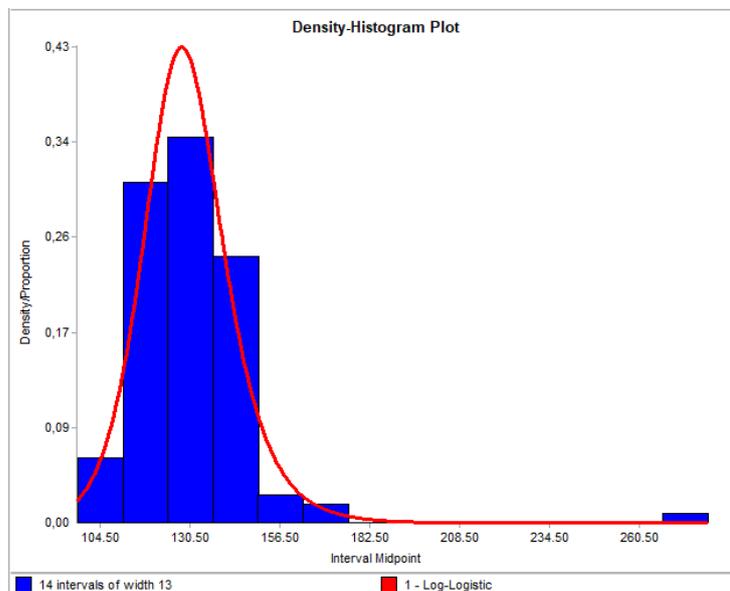
Fonte: Autor.

Figura 44 – Histograma da classe para o operador 3 de 0% a 25%



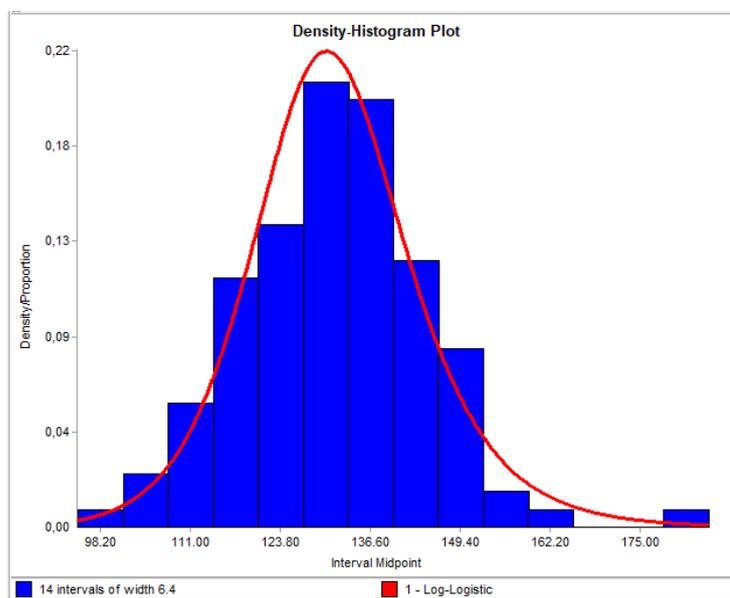
Fonte: Autor.

Figura 45 – Histograma da classe para o operador 3 de 25% a 50%



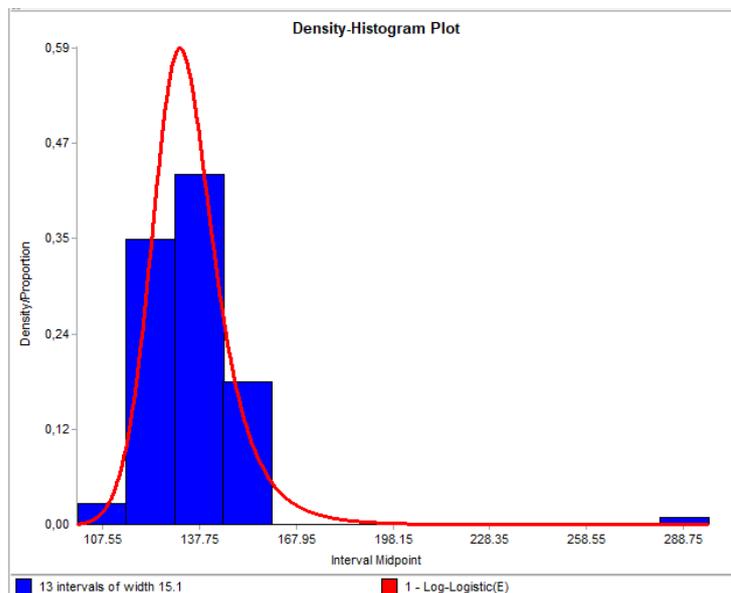
Fonte: Autor.

Figura 46 – Histograma da classe para o operador 3 de 50% a 75%



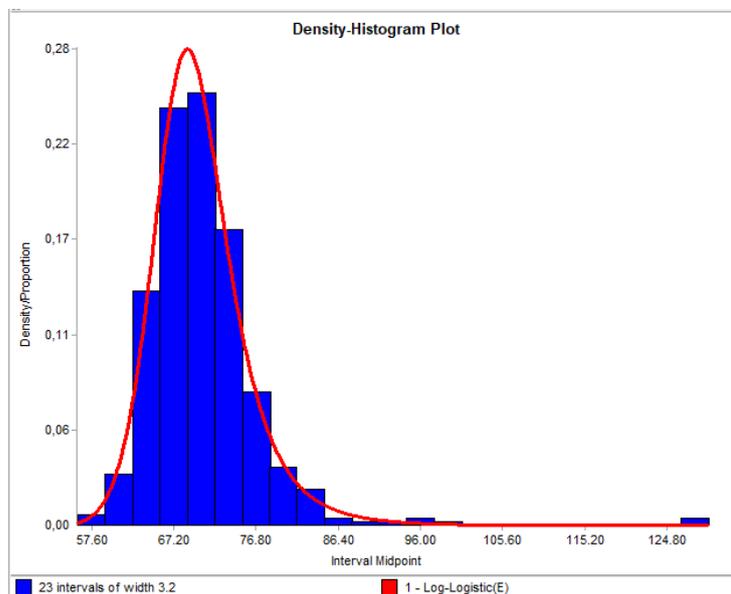
Fonte: Autor.

Figura 47 – Histograma da classe para o operador 3 de 75% a 100%



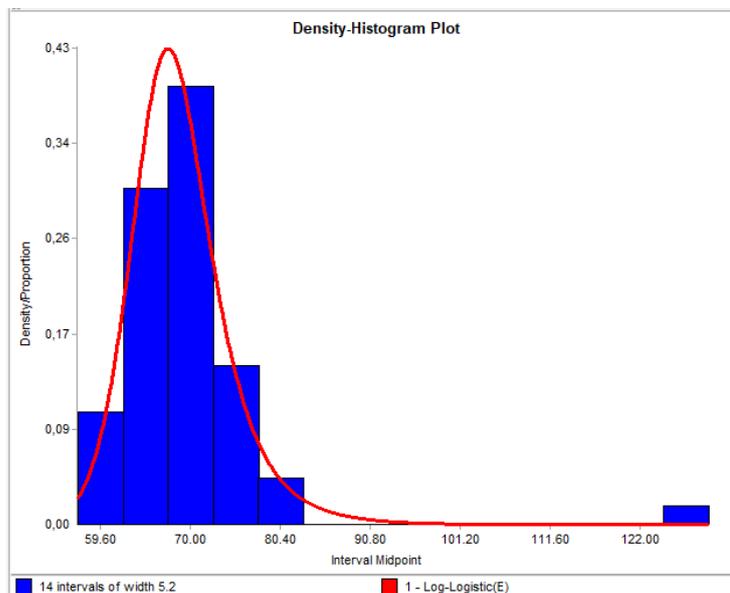
Fonte: Autor.

Figura 48 – Histograma da classe base para o operador 4



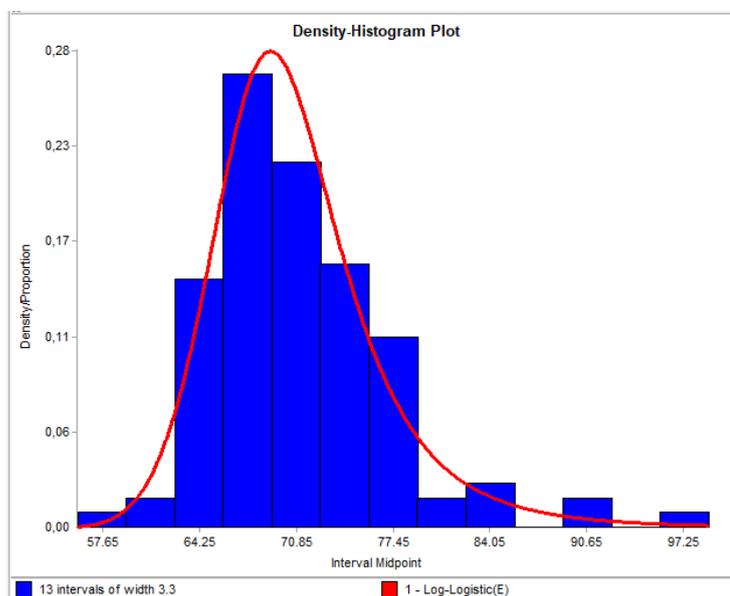
Fonte: Autor.

Figura 49 – Histograma da classe para o operador 4 de 0% a 25%



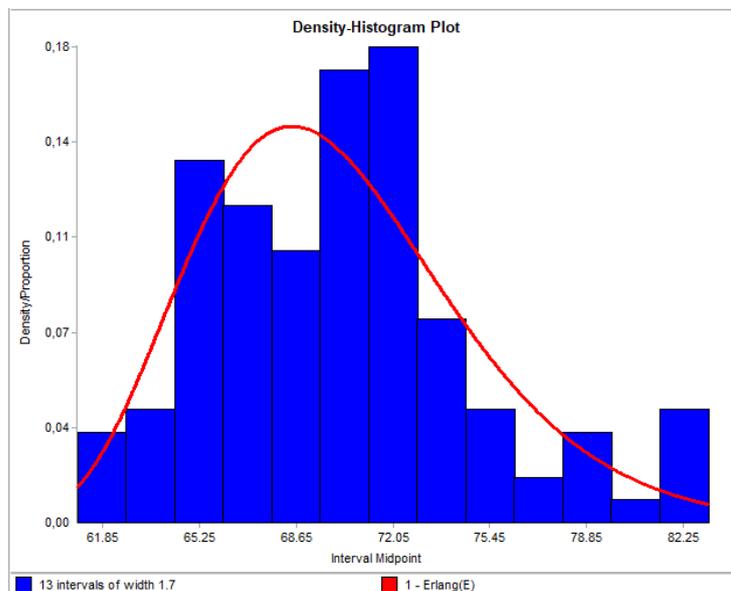
Fonte: Autor.

Figura 50 – Histograma da classe para o operador 4 de 25% a 50%



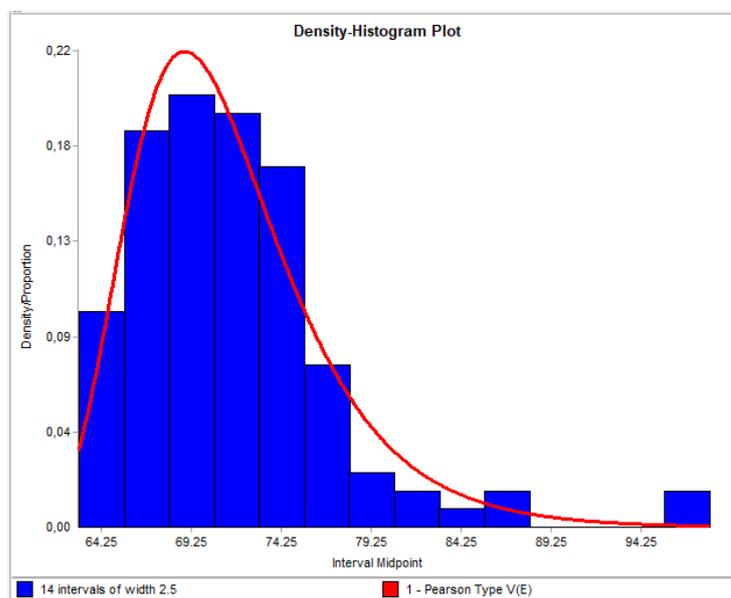
Fonte: Autor.

Figura 51 – Histograma da classe para o operador 4 de 50% a 75%



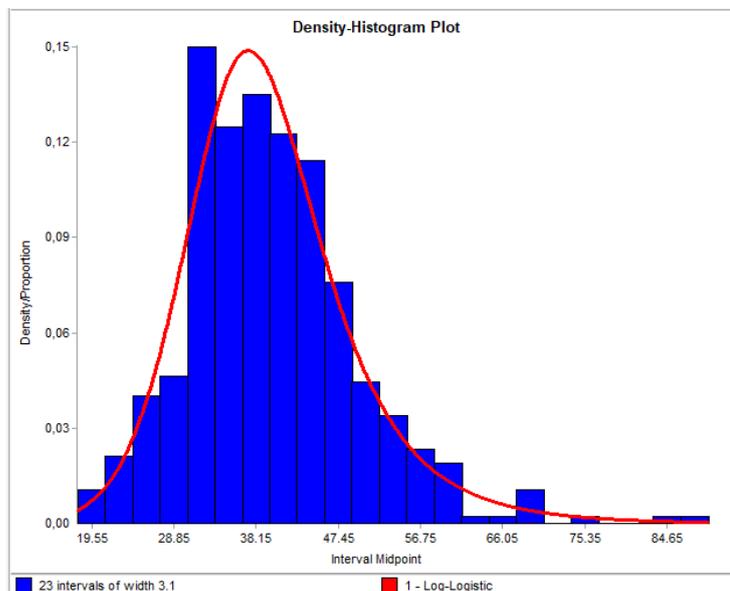
Fonte: Autor.

Figura 52 – Histograma da classe para o operador 4 de 75% a 100%



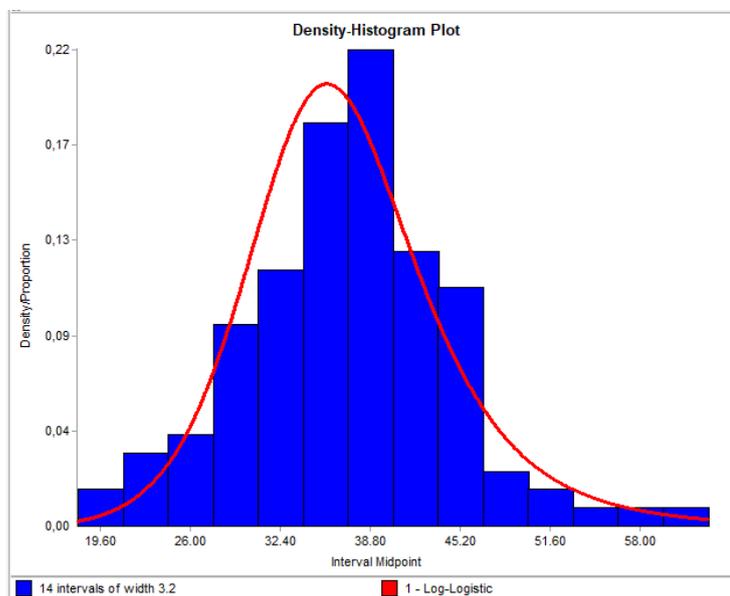
Fonte: Autor.

Figura 53 – Histograma da classe base para o operador 5



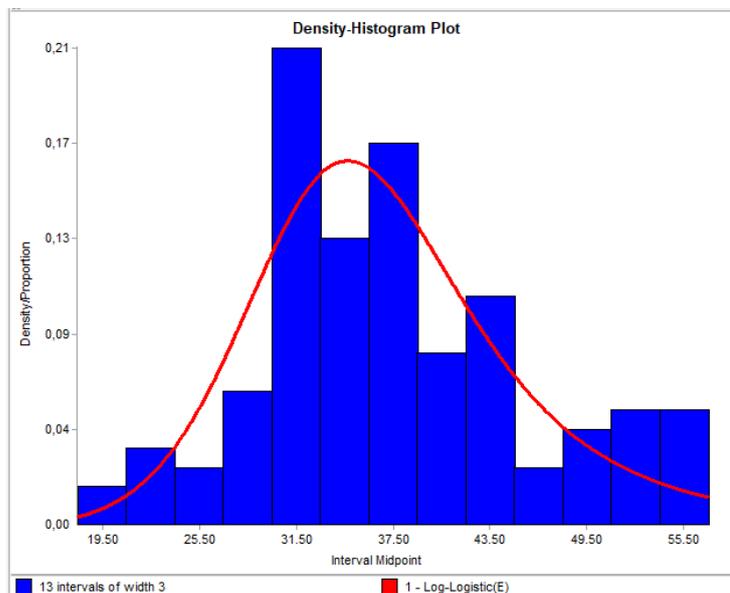
Fonte: Autor.

Figura 54 – Histograma da classe para o operador 5 de 0% a 25%



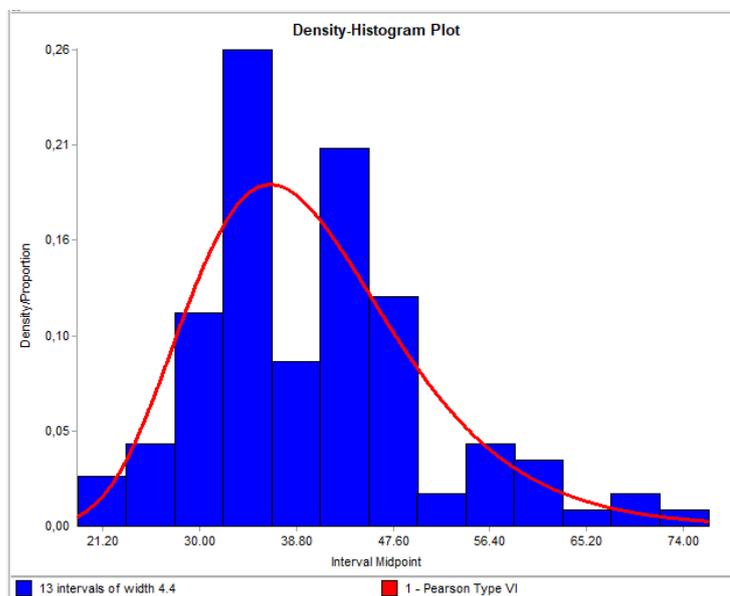
Fonte: Autor.

Figura 55 – Histograma da classe para o operador 5 de 25% a 50%



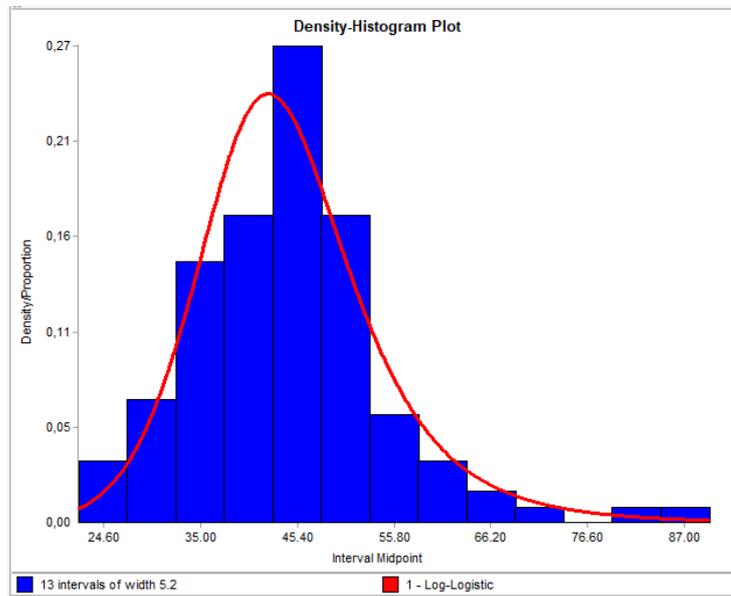
Fonte: Autor.

Figura 56 – Histograma da classe para o operador 5 de 50% a 75%



Fonte: Autor.

Figura 57 – Histograma da classe para o operador 5 de 75% a 100%



Fonte: Autor.

Anexos

ANEXO A – SIMBOLOGIA DO IDEF-SIM

Tabela 13 – Simbologia do IDEF-SIM

Elementos	Símbolo	Origem
Entidade		IDEF3 (modo de descrição de transições)
Função		IDEF0
Fluxo de Entidade		IDEF0 e IDEF3
Recurso		IDEF0
Controle		IDEF0
Regras para fluxos alternativos e/ou paralelos	 Regra E	IDEF3
	 Regra OU	
	 Regra E/OU	
Movimentação		Fluxograma
Comentário		IDEF0 e IDEF3
Fluxo de entrada em sistema modelado		
Ponto final do sistema		
Conexão com outra figura		
Portão de bloqueio		
Portão de liberação		

Fonte: Adaptado de Leal, Almeida e Montevechi (2008), Montevechi *et al.* (2010), Rocha (2020) e Amaral (2021)

ANEXO B – COMANDOS FFMPEG

```
mkdir -p frames
ffmpeg -i take12.mp4 -r 5 ./frames/take12_%05d.bmp
ffmpeg -i take13.mp4 -r 5 ./frames/take13_%05d.bmp
ffmpeg -i take14.mp4 -r 5 ./frames/take14_%05d.bmp
ffmpeg -i take15.mp4 -r 5 ./frames/take15_%05d.bmp
ffmpeg -i take16.mp4 -r 5 ./frames/take16_%05d.bmp
ffmpeg -i take17.mp4 -r 5 ./frames/take17_%05d.bmp
ffmpeg -i take18.mp4 -r 5 ./frames/take18_%05d.bmp
ffmpeg -i take19.mp4 -r 5 ./frames/take19_%05d.bmp
ffmpeg -i take20.mp4 -r 5 ./frames/take20_%05d.bmp
ffmpeg -i take21.mp4 -r 5 ./frames/take21_%05d.bmp
ffmpeg -i take22.mp4 -r 5 ./frames/take22_%05d.bmp
ffmpeg -i take23.mp4 -r 5 ./frames/take23_%05d.bmp
ffmpeg -i take24.mp4 -r 5 ./frames/take24_%05d.bmp
```

ANEXO C – SCRIPT PARA DETERMINAR PONTOS CENTRAIS

```

spark.read.option("header", "true").csv("centerPoints.csv").selectExpr("
    ↪ int(take) as take", "int(operador) as operador", "int(x) as x", "
    ↪ int(y) as y").createOrReplaceTempView("centerPoints")
spark.read.option("delimiter", "|").csv("emotids.txt").selectExpr("_c0 as
    ↪ file", "int(_c1) as x1", "int(_c2) as y1", "int(_c3) as x2", "
    ↪ int(_c4) as y2").selectExpr("file", "x1", "y1", "x2", "y2", "(x1+
    ↪ x2)/2 as cx", "(y1+y2)/2 as cy").createOrReplaceTempView("
    ↪ emotids")

spark.sql("select *, sqrt(pow(cx-x,2)+pow(cy-y,2)) as distance
    ↪ from emotids left outer join centerPoints on emotids.file like
    ↪ concat('take', centerPoints.take, '%')").createOrReplaceTempView("
    ↪ centerDistances")

spark.sql("select * from centerDistances").show()

spark.sql("select * from centerDistances order by file asc, operador asc,
    ↪ distance asc").createOrReplaceTempView("centerDistancesOrdered")

spark.sql("select * from centerDistancesOrdered").show()

spark.sql("select file, operador, cx, cy, element_at(collect_list(x1), 1)
    ↪ as x1, element_at(collect_list(x2), 1) as x2, element_at(
    ↪ collect_list(y1), 1) as y1, element_at(collect_list(y2), 1) as y2,
    ↪ element_at(collect_list(distance), 1) as distance from
    ↪ centerDistancesOrdered where x >= x1 and x <= x2 and y >= y1 and y
    ↪ <= y2 group by file, operador, cx, cy").createOrReplaceTempView("
    ↪ centerDistancesSelected")

spark.sql("select * from centerDistancesSelected").write.json("out/
    ↪ centerDistancesSelected")

```

```
spark.read.json("out/centerDistancesSelected/part-*.json").
```

```
  ↪ createOrReplaceTempView("operatorRectangles")
```

```
spark.sql("select concat(file,'_',cx,'_',cy,'_',x1,'_',y1,'_',x2,'_',y2) from
```

```
  ↪ operatorRectangles").coalesce(1).write.text("out/emoticInput")
```

```
spark.sql("select file, collect_list(struct(operator, cx, cy, x1, y1, x2,
```

```
  ↪ y2)) as rectangles from operatorRectangles group by file").
```

```
  ↪ createOrReplaceTempView("groupedRectangles")
```

```
spark.sql("select row_number() over (order by file) as row, * from
```

```
  ↪ groupedRectangles where size(rectangles) > 0").
```

```
  ↪ createOrReplaceTempView("groupedRectanglesRow")
```

```
spark.sql("select * from groupedRectanglesRow where row % 1000 = 0").
```

```
  ↪ coalesce(1).write.json("out/rectangleSamples")
```

ANEXO D – SCRIPT PARA CRUZAMENTO DE DADOS

```

import org.apache.spark.sql.{Row, Dataset}
import org.apache.spark.sql.types.{StructType, StructField, StringType,
  ↪ IntegerType, LongType, DoubleType, ArrayType}
import org.apache.spark.sql.catalyst.expressions.GenericRowWithSchema

// from https://gist.github.com/zouzias/44723de11222535223fe59b4b0bc228c
def addIndex(df: Dataset[Row], columnName: String = "idx") : Dataset[Row]
  ↪ = {
  val newSchema = StructType(df.schema.fields ++ Array(StructField(
    ↪ columnName, LongType, false)))
  val rddWithId = df.rdd.zipWithIndex
  val dfZippedWithId = spark.createDataFrame(rddWithId.map{ case (row,
    ↪ index) => Row.fromSeq(row.toSeq ++ Array(index))}, newSchema)
  return dfZippedWithId
}

def getOperacoes(eventos: Dataset[Row]) : Dataset[Row] = {
  val newSchema = StructType(eventos.schema.fields ++ Array(StructField(
    ↪ "operacao", IntegerType, true)))

  var currentTake: Int = 0
  var currentOperador: Int = 0
  var currentTrigger: Int = 2
  var currentFrame: Int = 0

  var nextOperacao: Int = 1
  var operacao: Integer = null

  // TODO: remove duration unused

```

```
var duration: Integer = 0

val operacoesRdd = eventos.coalesce(1).rdd.map(row => {
    val take: Int = row.getAs("take")
    val operador: Int = row.getAs("operador")
    val trigger: Int = row.getAs("trigger")
    val frame: Int = row.getAs("frame")
    val file: String = row.getAs("file")

    // Check if new sequence is starting
    if(take != currentTake || operador != currentOperador) {
        currentTake = take
        currentOperador = operador

        if(currentTrigger != 2) {
            throw new Exception(s"Sequence finished with ${
                ↪ currentTrigger} on ${file}")
        }

        nextOperacao = 1
        currentTrigger = 2
        currentFrame = 0
        duration = 0
    }

    if(trigger == 1 && currentTrigger != 2) {
        throw new Exception(s"Process started without finishing the
            ↪ previous one on ${file}")
    }

    if((trigger == 2 || trigger == 3) && currentTrigger != 1 &&
        ↪ currentTrigger != 3 && currentTrigger != 5) {
```

```
        throw new Exception(s"Process finished without starting on ${
            ↪ file}")
    }

    if(trigger == 4 && currentTrigger != 1 && currentTrigger != 3 &&
        ↪ currentTrigger != 5) {
        throw new Exception(s"Process paused without starting on ${
            ↪ file}")
    }

    if(trigger == 5 && currentTrigger != 4) {
        throw new Exception(s"Process unpausing without pausing on ${
            ↪ file}")
    }

    if(trigger == 1 || trigger == 3) {
        operacao = nextOperacao
        nextOperacao += 1
    }

    if(trigger == 2 || trigger == 3 || trigger == 4) {
        duration += frame - currentFrame
    }

    currentTrigger = trigger
    currentFrame = frame

    val newRow = Row.fromSeq(Array(take, frame, trigger, operador,
        ↪ file, operacao))

    if(trigger == 2) {
        operacao = null
    }
}
```

```

        duration = 0
    }

    newRow
})

return spark.createDataFrame(operacoesRdd, newSchema)
}

val rawMoods = (12 to 24).map((i) => {
    val inputFile = s"mood/take${i}.txt"
    val outputFile = s"mood/take${i}_mood.txt"

    addIndex(spark.read.option("delimiter", "\t").csv(inputFile).
        selectExpr("element_at(split(_c0,'/'),_c-1) as file", "int(_c1) as
            ↪ x1", "int(_c2) as y1", "int(_c3) as x2", "int(_c4) as y2"))
        ↪ .
        createOrReplaceTempView(s"take${i}_mood_input")

    val outputSchema = StructType(Array(StructField("file", StringType),
        ↪ StructField("valence", DoubleType), StructField("arousal",
        ↪ DoubleType), StructField("dominance", DoubleType), StructField(
        ↪ "discrete", ArrayType(StringType))))

    val outputRdd = spark.sparkContext.textFile(outputFile).map((line) =>
        ↪ {
        val cols = line.split("\t")
        Row.fromSeq(Seq(cols(0).substring(cols(0).lastIndexOf('/') + 1),
            ↪ cols(cols.length - 3).toDouble, cols(cols.length - 2).
            ↪ toDouble, cols(cols.length - 1).toDouble, cols.slice(1, cols
            ↪ .length - 3)))
    })
})

```

```

addIndex(spark.createDataFrame(outputRdd, outputSchema)).
  ↪ createOrReplaceTempView(s"take${i}_mood_output")

spark.sql(s"select i.*, valence, arousal, dominance, discrete from
  ↪ take${i}_mood_input as i full outer join take${i}_mood_output
  ↪ as o on i.idx = o.idx")
}).reduce((a, b) => a.union(b))
rawMoods.createOrReplaceTempView("rawMoods")

// affection, anger, annoyance, anticipation, aversion, confidence, disapproval,
  ↪ disconnection, disquietment, doubt_confusion, embarrassment, engagement
  ↪ , esteem, excitement, fatigue, fear, happiness, pain, peace, pleasure,
  ↪ sadness, sensitivity, suffering, surprise, sympathy, yearning
val discreteCategories = Seq("Affection", "Anger", "Annoyance", "
  ↪ Anticipation", "Aversion", "Confidence", "Disapproval", "
  ↪ Disconnection",
  "Disquietment", "Doubt/Confusion", "Embarrassment", "Engagement", "
  ↪ Esteem", "Excitement", "Fatigue", "Fear",
  "Happiness", "Pain", "Peace", "Pleasure", "Sadness", "Sensitivity",
  ↪ "Suffering", "Surprise", "Sympathy", "Yearning")
val discreteColumns = discreteCategories.map((category) => category.
  ↪ toLowerCase().replaceAll("\\W", "_"))
val selectCategories = discreteCategories.map((category) => s"
  ↪ array_contains(discrete, '${category}') as ${category.toLowerCase().
  ↪ replaceAll("\\W", "_")}").mkString(", ")
spark.sql(s"select *, ${selectCategories} from rawMoods").
  ↪ createOrReplaceTempView("moods")
val avgCategories = discreteColumns.map((category) => s"avg(int(${
  ↪ category})) as ${category}").mkString(", ")
spark.sql(s"select count(*), ${avgCategories} from moods").show()

```

```

spark.read.option("header", "true").csv("pessoas.csv").
    ↪ createOrReplaceTempView("pessoas")
spark.sql("select_*_from_pessoas").show()
spark.read.option("header", "true").csv("eventos.csv").
    selectExpr("int(Take)_as_take",
        "int(Frame)_as_frame",
        "int(Trigger)_as_trigger",
        "int(Operador)_as_operador",
        "concat('take',_take,_'_',_lpad(frame,_5,_'0'))_as_file").
    createOrReplaceTempView("eventos")

val eventos = spark.sql("select_*_from_eventos_order_by_take,_operador,_
    ↪ frame")
addIndex(eventos).createOrReplaceTempView("eventosIdx")
println("Erros_em_eventos:")
val errors = spark.sql("select_current.take,_current.operador,_last.
    ↪ trigger_as_lastTrigger,_current.trigger,_last.frame_as_lastFrame,_
    ↪ current.frame_from_eventosIdx_current_left_outer_join_eventosIdx_
    ↪ last_on_current.idx=_last.idx+_1_and_current.take=_last.take_and
    ↪ _current.operador=_last.operador_where_(current.trigger=_1_and_
    ↪ last.trigger_IS_NOT_NULL_and_last.trigger_<>_2)_or_(current.trigger
    ↪ _=_2_and_(last.trigger_IS_NULL_or_last.trigger=_2_or_last.trigger_
    ↪ =_4))_or_(current.trigger=_3_and_last.trigger_<>_1_and_last.
    ↪ trigger_<>_3_and_last.trigger_<>_5)_or_(current.trigger=_4_and_(
    ↪ last.trigger_IS_NULL_or_last.trigger=_2_or_last.trigger=_4))_or_(
    ↪ current.trigger=_5_and_last.trigger_<>_4)_order_by_current.take,_
    ↪ current.operador,_current.frame")
errors.show(1000)
//errors.write.option("header", "true").csv("out/errors")

addIndex(getOperacoes(eventos)).createOrReplaceTempView("eventosOperacoes
    ↪ ")

```

```

spark.sql("select first.*, second.trigger as nextTrigger, second.frame-
↳ first.frame as duration from eventosOperacoes first left outer join
↳ eventosOperacoes second on second.idx=first.idx+1 and first.
↳ take=second.take and first.operador=second.operador order by
↳ first.idx").
    createOrReplaceTempView("eventosOperacoesDuration")

val operacoesRdd = spark.sql("select take, operador, operacao,
↳ collect_list(struct(idx, frame, trigger, nextTrigger, duration)) as
↳ sequence from eventosOperacoesDuration group by take, operador,
↳ operacao").
    rdd.map(row => {
        val take: Int = row.getAs("take")
        val operador: Int = row.getAs("operador")
        val operacao: Int = row.getAs("operacao")
        val sequence: Seq[GenericRowWithSchema] = row.getAs("sequence")
        val frame: Integer = sequence(0).getAs("frame")

        var duration: Double = 0.0
        var durationProcess: Double = 0.0

        for(a <- 0 until sequence.length) {
            val entry = sequence(a)
            val trigger: Integer = entry.getAs("trigger")
            val entryDuration: Integer = entry.getAs("duration")

            if(trigger != 2) {
                duration += entryDuration

                if(trigger != 4) {
                    durationProcess += entryDuration
                }
            }
        }
    })

```

```

    }
  }

  Row.fromSeq(Array(take, operador, operacao, frame, frame +
    ↪ duration.toInt, duration / 5, durationProcess / 5))
})

val operacoesSchema = StructType(Array(StructField("take", IntegerType,
  ↪ false),
  StructField("operador", IntegerType, false),
  StructField("operacao", IntegerType, false),
  StructField("firstFrame", IntegerType, false),
  StructField("finalFrame", IntegerType, false),
  StructField("duration", DoubleType, false),
  StructField("durationProcess", DoubleType, false)))

spark.createDataFrame(operacoesRdd, operacoesSchema).
  ↪ createOrReplaceTempView("operacoes")
// spark.sql("select operacoes.*, pessoa from operacoes left outer join
  ↪ pessoas on operacoes.take = pessoas.take and operacoes.operador =
  ↪ pessoas.operador").
// createOrReplaceTempView("operacoesPessoa")
// spark.sql("select * from operacoesPessoa").show()
spark.sql("select take, operador, avg(duration) as durationAvg, stddev(
  ↪ duration) as durationStddev from operacoes group by take, operador
  ↪ order by operador, take").createOrReplaceTempView("durationAvg")
spark.sql("select * from durationAvg").show(1000)

val outliers = spark.sql("select operacoes.*, durationAvg, durationStddev
  ↪ from operacoes left outer join durationAvg on operacoes.take =
  ↪ durationAvg.take and operacoes.operador = durationAvg.operador
  ↪ where duration > durationAvg + 2 * durationStddev or duration <
  ↪ durationAvg - 2 * durationStddev")

```

```

outliers.write.option("header", "true").csv("out/outliers")

// load selected rectangles
spark.read.json("out/centerDistancesSelected/part-*.json").selectExpr("*",
    ↪ "int(substring(file, 5, 2)) as take", "int(substring(file, 8, 5)) as
    ↪ frame").createOrReplaceTempView("operatorRectangles")
spark.sql("select * from operatorRectangles").show()

spark.sql("select operatorRectangles.*, pessoa from operatorRectangles
    ↪ left outer join pessoas on operatorRectangles.take = pessoas.take
    ↪ and operatorRectangles.operator = pessoas.operator and
    ↪ operatorRectangles.operator = pessoas.operator").
    createOrReplaceTempView("operadores")
spark.sql("select * from operadores").show()

spark.sql("select pessoa, operador, take, frame, moods.* from operadores
    ↪ left outer join moods on operadores.file = moods.file and
    ↪ operadores.x1 = moods.x1 and operadores.y1 = moods.y1 and
    ↪ operadores.x2 = moods.x2 and operadores.y2 = moods.y2").drop("idx",
    ↪ "x1", "y1", "x2", "y2").
    createOrReplaceTempView("operadoresMood")

spark.sql("select operadoresMood.*, operacao, firstFrame, finalFrame,
    ↪ duration, durationProcess from operadoresMood left outer join
    ↪ operacoes on operadoresMood.take = operacoes.take and
    ↪ operadoresMood.operator = operacoes.operator and operadoresMood.
    ↪ frame >= operacoes.firstFrame and operadoresMood.frame <= operacoes.
    ↪ finalFrame").
    createOrReplaceTempView("operadoresOperacao")

spark.sql(s"select take, operador, pessoa, operacao, firstFrame,
    ↪ finalFrame, duration, durationProcess, count(*) as frames, avg(

```

```

    ↪ valence) as valence, avg(arousal) as arousal, avg(dominance) as
    ↪ dominance, ${avgCategories} from operadoresOperacao group by take,
    ↪ operador, pessoa, operacao, firstFrame, finalFrame, duration,
    ↪ durationProcess").
    createOrReplaceTempView("operacoesMood")
spark.sql("select * from operacoesMood").coalesce(1).write.option("header
    ↪ ", "true").csv("out/processos")

spark.sql(s"select take, operador, pessoa, avg(valence) as valence, avg(
    ↪ arousal) as arousal, avg(dominance) as dominance, ${discreteColumns.
    ↪ map((column) => s"avg(${column}) as ${column}") .mkString(", ")}
    ↪ from operacoesMood group by take, operador, pessoa order by
    ↪ operador, take").
    createOrReplaceTempView("operadorAvg")
spark.sql("select * from operadorAvg").
    coalesce(1).write.option("header", "true").csv("out/operadorAvg")

```

ANEXO E – *SCRIPT* PARA VALIDAÇÃO DO PONTO CENTRAL

E.1 Arquivo `draw_operator_samples.py`

```
import pathlib
import sys
import json

from PIL import Image
from PIL import ImageFont

from utils import draw_bounding_box
from utils import draw_point

if len(sys.argv) < 4:
    exit('numero de argumentos')

images_dir = pathlib.Path(sys.argv[1])
output_dir = pathlib.Path(sys.argv[2])

samples_file = pathlib.Path(sys.argv[3])

colors = ['#0000ff', '#00ff00', '#ff0000', '#00ffff', '#ff00ff', '#ffff00',
↪ ]
font = ImageFont.load_default()

def draw_sample(sample):
    print(sample['file'])
```

```
img_file = images_dir / sample["file"]
output_file = output_dir / sample["file"]
if img_file.exists():
    with Image.open(img_file) as im:
        for rectangle in sample["rectangles"]:
            draw_point(im, rectangle["cpx"], rectangle["cpy"], 10,
                ↪ colors[rectangle["operador"]])
            draw_bounding_box(
                im,
                rectangle["x1"],
                rectangle["x2"],
                rectangle["y1"],
                rectangle["y2"],
                colors[rectangle["operador"]],
                font,
                display_str_list=[f"operador_{rectangle['operador']}"])
        im.save(output_file)

with open(samples_file, "r") as samples:
    lines = samples.readlines()
    for line in lines:
        sample = json.loads(line)
        draw_sample(sample)
```

E.2 Arquivo utils.py

```
import numpy as np

from PIL import Image
from PIL import ImageColor
from PIL import ImageDraw
```

```
from PIL import ImageFont

def draw_point(image, x, y, size, color, thickness=4):
    draw = ImageDraw.Draw(image)
    draw.line((x - size, y, x + size, y), width=thickness, fill=color)
    draw.line((x, y - size, x, y + size), width=thickness, fill=color)

def draw_bounding_box(image,
                      left,
                      right,
                      top,
                      bottom,
                      color,
                      font,
                      thickness=4,
                      display_str_list=()):
    """Adds a bounding box to an image."""
    draw = ImageDraw.Draw(image)
    draw.line([(left, top), (left, bottom), (right, bottom), (right, top),
              (left, top)],
              width=thickness,
              fill=color)

    # If the total height of the display strings added to the top of the
    #   ↪ bounding
    # box exceeds the top of the image, stack the strings below the
    #   ↪ bounding box
    # instead of above.
    display_str_heights = [font.getsize(ds)[1] for ds in display_str_list]
    # Each display_str has a top and bottom margin of 0.05x.
    total_display_str_height = (1 + 2 * 0.05) * sum(display_str_heights)
```

```
if top > total_display_str_height:
    text_bottom = top
else:
    text_bottom = top + total_display_str_height
# Reverse list and print from bottom to top.
for display_str in display_str_list[::-1]:
    text_width, text_height = font.getsize(display_str)
    margin = np.ceil(0.05 * text_height)
    draw.rectangle([(left, text_bottom - text_height - 2 * margin),
                    (left + text_width, text_bottom)],
                    fill=color)
    draw.text((left + margin, text_bottom - text_height - margin),
              display_str,
              fill="black",
              font=font)
    text_bottom -= text_height - 2 * margin

def draw_bounding_box_on_image(image,
                               ymin,
                               xmin,
                               ymax,
                               xmax,
                               color,
                               font,
                               thickness=4,
                               display_str_list=()):
    """Adds a bounding box to an image."""
    im_width, im_height = image.size
    (left, right, top, bottom) = (xmin * im_width, xmax * im_width,
                                   ymin * im_height, ymax * im_height)
    draw_bounding_box(left, right, top, bottom, color, font, thickness=
        ↪ thickness, display_str_list=display_str_list)
```

```
def draw_boxes(image, boxes, class_names, scores, max_boxes=10, min_score
    ↪ =0.1):
    """Overlay labeled boxes on an image with formatted scores and label
    ↪ names."""
    colors = list(ImageColor.colormap.values())

    try:
        font = ImageFont.truetype("/usr/share/fonts/truetype/liberation/
            ↪ LiberationSansNarrow-Regular.ttf",
                                   25)

    except IOError:
        print("Font not found, using default font.")
        font = ImageFont.load_default()

    for i in range(min(boxes.shape[0], max_boxes)):
        class_name = class_names[i].numpy().decode("ascii")
        if scores[i] >= min_score:
            ymin, xmin, ymax, xmax = tuple(boxes[i])
            display_str = "{}: {}%".format(class_name,
                                           int(100 * scores[i]))
            color = colors[hash(class_name) % len(colors)]
            image_pil = Image.fromarray(np.uint8(image)).convert("RGB")
            draw_bounding_box_on_image(
                image_pil,
                ymin,
                xmin,
                ymax,
                xmax,
                color,
                font,
```

```
        display_str_list=[display_str])
    np.copyto(image, np.array(image_pil))
return image
```

E.3 Arquivo requirements.txt

```
absl-py==0.13.0
astunparse==1.6.3
cached-property==1.5.2
cachetools==4.2.2
certifi==2021.5.30
chardet==4.0.0
dataclasses==0.8
flatbuffers==1.12
gast==0.4.0
google-auth==1.32.0
google-auth-oauthlib==0.4.4
google-pasta==0.2.0
grpcio==1.34.1
h5py==3.1.0
idna==2.10
importlib-metadata==4.6.0
keras-nightly==2.5.0.dev2021032900
Keras-Preprocessing==1.1.2
Markdown==3.3.4
numpy==1.19.5
oauthlib==3.1.1
opt-einsum==3.3.0
Pillow==8.2.0
pkg-resources==0.0.0
protobuf==3.17.3
pyasn1==0.4.8
```

```
pyasn1-modules==0.2.8
requests==2.25.1
requests-oauthlib==1.3.0
rsa==4.7.2
six==1.15.0
tensorboard==2.5.0
tensorboard-data-server==0.6.1
tensorboard-plugin-wit==1.8.0
tensorflow==2.5.0
tensorflow-estimator==2.5.0
tensorflow-hub==0.12.0
termcolor==1.1.0
typing-extensions==3.7.4.3
urllib3==1.26.6
Werkzeug==2.0.1
wrapt==1.12.1
zipp==3.4.1
```

E.4 Comando para executar o *script*

```
python draw_operator_samples.py <caminho para a pasta contendo os quadros
  ↳ > <caminho para salvar os quadros> <caminho para o arquivo
  ↳ exportado com os retangulos de exemplo>
```

ANEXO F – SCRIPTS PARA CÁLCULO DA CORRELAÇÃO E SELEÇÃO DE VARIÁVEIS

F.1 Arquivo common.R

```
library("readr")
library("Hmisc")
library("rio")
library("gtools")
library("santoku")
library("stringr")
library("dplyr")
library("purrr")
library("ggplot2")
library("gridExtra")

library(gamlss)
library(gamlss.dist)
library(gamlss.add)

correlations <- function(processos) {
  cm <- rcorr(as.matrix(processos), type="spearman")
  corr.P <- as.data.frame(cm$P)
  corr.r <- as.data.frame(cm$r)
  corrOrder <- order(corr.P$durationProcess, -abs(corr.r$durationProcess))
  ↪
  corr.P <- corr.P[corrOrder,]
  corr.r <- corr.r[corrOrder,]
  return(list(r=corr.r, P=corr.P))
}

alternateSort <- function(corr) {
```

```
corrOrder <- order(-abs(corr$r$durationProcess), corr$p$durationProcess)
  ↪
P <- corr$p[corrOrder,]
r <- corr$r[corrOrder,]
return(list(r=r, P=P))
}

filterCorr <- function(corr) {
  r <- corr$r[corr$p$durationProcess <= 0.05,]
  P <- corr$p[corr$p$durationProcess <= 0.05,]
  return(list(r=r, P=P))
}

null2na <- function(x) {
  if(is.null(x)) {
    return(NA)
  }

  return(x)
}

getSubsets <- function(dist, q, op) {
  n <- length(q) - 1
  meta <- data.frame(q = integer(),
                    size = integer(),
                    start = character(),
                    end = character(),
                    startValue = numeric(),
                    endValue = numeric(),
                    stringsAsFactors = FALSE)

  sub <- list()
  i <- 1
```

```
previousEq <- F
while(i <= n) {
  start <- q[i]
  finish <- NULL
  for(j in (i+1):(n+1)) {
    finish <- q[j]
    if(j >= n | finish != q[j + 1]) {
      break
    }
  }
  row <- nrow(meta) + 1
  if(start == finish) {
    sub[[row]] <- op[dist == start,]
    previousEq <- T
  } else {
    if(previousEq & j == (n+1)) {
      sub[[row]] <- op[dist > start & dist <= finish ,]
    } else if(previousEq) {
      sub[[row]] <- op[dist > start & dist < finish ,]
    } else if(j == (n+1)) {
      sub[[row]] <- op[dist >= start & dist <= finish ,]
    } else {
      sub[[row]] <- op[dist >= start & dist < finish ,]
    }
    previousEq <- F
  }
  meta[row,] <- list(i,
                    nrow(sub[[row]]),
                    names(start),
                    names(finish),
                    start,
                    finish)
```

```
    i <- j
  }

  return(list(sub=sub, meta=meta))
}

bestFits <- function(processos, operador) {
  op <- processos[processos$operador == operador,]

  distributions <- data.frame(operador = numeric(),
                              size = integer(),
                              var = character(),
                              start = character(),
                              end = character(),
                              startValue = numeric(),
                              endValue = numeric(),
                              code = character(),
                              name = character(),
                              mu = numeric(),
                              sigma = numeric(),
                              nu = numeric(),
                              tau = numeric(),
                              stringsAsFactors = FALSE)

  distributionsIndex = 1;

  export(data.frame(durationProcess = op$durationProcess), str_interp("
    ↪ out/dist/op${operador}.xlsx"))
  export(data.frame(durationProcess = op$durationProcess), str_interp("
    ↪ out/dist/op${operador}.txt"), row.names = F, col.names = F)
  fit <- fitDist(op$durationProcess, k = 2, type = "realplus", trace =
    ↪ FALSE, try.gamlss = TRUE)
  #print(histDist(fit$y, family = fit$family[1], ylim = c(0, 0.05)))
}
```

```
distributions[distributionsIndex,] <- list(operador,
                                         nrow(op),
                                         NA,
                                         NA,
                                         NA,
                                         NA,
                                         NA,
                                         fit$family[1],
                                         fit$family[2],
                                         null2na(fit$mu.coefficients),
                                         null2na(fit$sigma.coefficients),
                                         null2na(fit$nu.coefficients),
                                         null2na(fit$tau.coefficients))

distributionsIndex <- distributionsIndex + 1

#histDist(fit$y, family = fit$family[1], ylim = c(0, 0.04))

corr.op <- filterCorr(correlations(op[, -c(1:5)]))
#View(corr.op$P)
selected <- row.names(corr.op$P)[1:3]

#for(attr in selected) {
# print(ggplot(op, aes(x = get(attr), y = durationProcess)) +
# geom_boxplot(aes(group = quantcut(get(attr), 4))) +
# labs(y = "Duracao do processo (s)", x = str_interp("${str_to_title(
#   ↪ attr}")))
#}

#corr.op2 <- alternateSort(corr.op)
#View(corr.op2$P)
#View(corr.op2$r)
#selected <- row.names(corr.op2$P)[1:3]
```

```

for(attr in selected) {
  dist <- get(attr, envir = as.environment(op))
  q <- quantile(dist)
  subsets <- getSubsets(dist, q, op)
  for(i in 1:(length(subsets$sub))) {
    sub <- subsets$sub[[i]]
    meta <- subsets$meta
    export(data.frame(durationProcess = sub$durationProcess), str_interp
      ↪ ("out/dist/op${operator}_${attr}_${meta$start[i]}_${meta$end[i]
      ↪ }).xlsx"))
    export(data.frame(durationProcess = sub$durationProcess), str_interp
      ↪ ("out/dist/op${operator}_${attr}_${meta$start[i]}_${meta$end[i]
      ↪ }).txt"), row.names = F, col.names = F)
    if(meta$size > 0) {
      fit <- fitDist(sub$durationProcess, k = 2, type = "realplus",
        trace = FALSE, try.gamlss = TRUE)
      #print(histDist(fit$y, family = fit$family[1], ylim = c(0, 0.05)))

      distributions[distributionsIndex,] <- list(operator,
        meta$size[i],
        attr,
        meta$start[i],
        meta$end[i],
        meta$startValue[i],
        meta$endValue[i],
        fit$family[1],
        fit$family[2],
        null2na(fit$mu.coefficients)
        ↪ ,
        null2na(fit$sigma.
        ↪ coefficients),

```

```

null2na(fit$nu.coefficients)
      ↪ ,
null2na(fit$tau.
      ↪ coefficients))
} else {
  distributions[distributionsIndex,] <- list(operador,
                                           0,
                                           attr,
                                           meta$start[i],
                                           meta$end[i],
                                           meta$startValue[i],
                                           meta$endValue[i],
                                           NA,
                                           NA,
                                           NA,
                                           NA,
                                           NA,
                                           NA)
}

distributionsIndex <- distributionsIndex + 1
}
}

return(distributions)
}

```

F.2 Arquivo correlation.R

```

source("common.R")

processos <- read_csv("processos.csv")[,-c(5, 6, 9)]

```

```
ocorrenciasDf <- read_csv("ocorrencias.csv")
# p3 <- processos[processos$peessoa == 3,]
# cm <- rcorr(as.matrix(p2[,c(3, 8:37)]))
# cm <- rcorr(as.matrix(p3))

corr <- correlations(processos[,-c(1:5)])
export(corr, "out/corr.xlsx", rowNames = T)

getCorrTable <- function(operador, corr) {
  variavel <- row.names(corr$P)
  correlacao <- corr$r[, "durationProcess"]
  pValue <- corr$P[, "durationProcess"]

  return(data.frame(operador, variavel, correlacao, pValue))
}

corrTable <- getCorrTable(0, corr)

for(operador in sort(unique(processos$operador))) {
  op <- processos[processos$operador == operador,]
  corr.op <- correlations(op[, -c(1:5)])
  corrTable <- rbind(corrTable, getCorrTable(operador, corr.op))
}

ocorrencias <- c()

for (i in 1:nrow(corrTable)) {
  row <- corrTable[i,]
  col <- match(row$variavel, names(ocorrenciasDf))

  if(row$operador == 0) {
    if(is.na(col)) {
```

```
    ocorrencias[[i]] <- sum(ocorrenciasDf$cnt)
  } else {
    ocorrencias[[i]] <- sum(ocorrenciasDf[, col])
  }
} else {
  if(is.na(col)) {
    ocorrencias[[i]] <- ocorrenciasDf[ocorrenciasDf$operador == row$
      ↪ operador,]$cnt
  } else {
    ocorrencias[[i]] <- ocorrenciasDf[ocorrenciasDf$operador == row$
      ↪ operador,][[col]]
  }
}
}

strongCorr <- corrTable[corrTable$pValue <= 0.05,]

corrTable$correlacao <- round(corrTable$correlacao, digits = 5)
corrTable$pValue <- format(corrTable$pValue, format = "e", digits = 4)
corrTable$ocorrencias <- format(ocorrencias, big.mark=",")

corrTable = corrTable[corrTable$variavel != "durationProcess",]

unique(strongCorr$variavel)

export(corrTable, "out/corrTable.csv")

# Plots:
boxPlot <- function(op, attr) {
  groups <- quantcut(get(attr, envir = as.environment(op)), 4)
  op$grp <- groups
}
```

```

op_mean <- op %>%
  group_by(grp) %>%
  summarize(average = mean(durationProcess),
            center = (min(get(attr)) + max(get(attr))) / 2 ) %>%
  ungroup()

plot <- ggplot(op, aes(x = get(attr), y = durationProcess)) +
  geom_boxplot(aes(group = groups)) +
  geom_point(data = op_mean,
            mapping = aes(x = center, y = average),
            size = 2,
            color="#AB3428") +
  geom_line(data = op_mean,
            mapping = aes(x = center, y = average, group=1),
            size = 1,
            color="#AB3428") +
  ggtitle(str_interp("Operador_{operator}")) +
  xlab(capitalize(attr)) + ylab("Tempo_{de}_{operacao}")
return(plot)
}

plots <- list()
for(operator in sort(unique(processos$operator))) {
  op <- processos[processos$operator == operator,]
  op <- op[!is.na(op$durationProcess),]
  corr.op <- filterCorr(correlations(op[, -c(1:5)]))
  attr <- row.names(corr.op$P)[1]

  plot <- boxPlot(op, attr)
  print(plot)

  plots[[operator]] <- plot
}

```

```

}

for(operador in 1:(length(plots))) {
  plot <- plots[[operador]]
  ggsave(str_interp("out/boxplots/operador${operador}.png"), plot = plot)
}

for(operador in sort(unique(processos$operador))) {
  op <- processos[processos$operador == operador,]
  op <- op[!is.na(op$durationProcess),]
  corr.op <- filterCorr(correlations(op[, -c(1:5)]))
  attrs <- row.names(corr.op$P[!is.na(corr.op$P$durationProcess),])
  for(i in 1:(length(attrs))) {
    attr <- attrs[[i]]
    plot <- boxPlot(op, attr)
    ggsave(str_interp("out/boxplots/all/operador${operador}_${i}_${attr}.
      ↪ png"), plot = plot)
  }
}

```

F.3 Arquivo fit.R

```

source("common.R")

processos <- read_csv("processos.csv")[,-c(5, 6, 9)]

fitTable <- NULL
for(operador in sort(unique(processos$operador))) {
  dists <- bestFits(processos, operador)
  if(is.null(fitTable)) {
    fitTable <- dists
  } else {

```

```
    fitTable <- rbind(fitTable, dists)
  }
}

export(fitTable, "out/fit.xlsx")
```