PONTIFÍCIA UNIVERSIDADE CATÓLICA
DO RIO DE JANEIRO


**SÉRGIO LUIS ALVES DELFINO**

**YASMIN SILVA AMARO DE LIMA**

# Ciência de Dados aplicada à análise do processo de vacinação contra a COVID-19 nos municípios do Brasil: Um estudo de caso do projeto ICODA\EFFECT-Brazil

PROJETO DE GRADUAÇÃO EM ENGENHARIA DE PRODUÇÃO
APRESENTADO AO DEPARTAMENTO DE ENGENHARIA INDUSTRIAL
DA PUC-RIO, COMO PARTE DOS REQUISITOS PARA OBTENÇÃO
DO TÍTULO DE ENGENHEIRO DE PRODUÇÃO

Orientadora: Fernanda Araujo Baião
Coorientador: Leonardo dos Santos Lourenço Bastos

Departamento de Engenharia Industrial
Rio de Janeiro, 17 de Novembro de 2021

# AGRADECIMENTOS

À Divina Providência.

## RESUMO

A COVID-19 surgiu em dezembro de 2019 através de um vírus desconhecido e de consequências inesperadas, ocasionando diversos óbitos pelo mundo. Para enfrentá-lo, diversas entidades tomaram medidas em nível individual e global. Dentre essas medidas, destacou-se o rápido desenvolvimento de vacinas para imunizar as populações. Especificamente no território brasileiro, todo o processo de vacinação foi monitorado pelo Sistema Único de Saúde através do Sistema de Informações do Programa Nacional de Imunizações (SIPNI), o que gerou um enorme volume de dados. Utilizar Ciência de Dados para analisar esses dados mostrou-se como uma metodologia promissora para trazer algum conhecimento útil e previamente desconhecido a respeito da COVID-19 e do impacto da vacinação na população. Tal metodologia utiliza habilidades estatísticas, computacionais e de domínio para gerar informações a partir de gráficos, classificações e previsões. Nesse sentido, o projeto ICODA (*International COVID-19 Data Alliance*) surge como uma iniciativa de colaboração global para responder à pandemia utilizando o poder dos dados. Como parte desse projeto, o presente trabalho se propôs a analisar os impactos da Campanha de Vacinação contra a COVID-19 no número de óbitos registrados nos municípios brasileiros. Dentro da aplicação do Ciclo de Vida de Ciência de Dados, especificamente na etapa de modelagem, percebemos que o modelo de Regressão de Poisson é o que melhor explica o número de óbitos ocorridos em vários municípios sete semanas após a vacinação. Após aplicar o modelo de Regressão de Poisson foi possível concluir que a redução do número de óbitos pode ser explicada pelo avanço da cobertura vacinal.



## PALAVRAS CHAVE

Ciência de Dados, COVID-19, Python, impacto da vacinação, regressão de Poisson

## ABSTRACT

COVID-19 appeared in December 2019 through an unknown virus and unexpected consequences, causing several deaths around the world. To face it, several entities have taken measures at the individual and global levels. Among these measures, the rapid development of vaccines to immunize populations stood out. Specifically in the Brazilian territory, the entire vaccination process was monitored by the Unified Health System through the Information System of the National Immunization Program (SIPNI), which generated a huge volume of data. Using Data Science to analyze these data has proven to be a promising methodology to bring some useful and previously unknown knowledge about COVID-19 and the impact of vaccination on the population. Such methodology uses statistical, computational and domain skills to generate information from graphs, rankings and predictions. In this sense, the ICODA project (International COVID Data Alliance) emerges as a global collaboration initiative to respond to the pandemic using the power of data. As part of this project, the present work aimed to analyze the impacts of the Vaccination Campaign against COVID-19 on the number of deaths registered in Brazilian municipalities. Within the application of the Data Science Life Cycle, specifically in the modeling stage, we realized that the Poisson Regression model is the one that best explains the number of deaths that occurred in several cities seven weeks after vaccination. After applying the Poisson Regression model, it was possible to conclude that the reduction in the number of deaths can be explained by the advance in vaccination coverage.

## KEYWORDS

Data Science, Covid-19, Python, vaccination impact, Poisson regression

# SUMÁRIO

# LISTA DE FIGURAS

# LISTA DE TABELAS

# 1 INTRODUÇÃO

Em Dezembro de 2019 o mundo foi impactado pelo Coronavírus, um vírus desconhecido que se espalha pelo ar e de reações inesperadas. Com o primeiro relato de aparição na cidade de Wuhan, na China, se espalhou para os demais países com uma velocidade inesperada, fazendo com que governos e entidades de saúde globais tomassem medidas para conter seu avanço [Croda e Garcia, 2020]. Não conhecer o vírus e as consequências da Coronavírus Disease - 19, ou Covid-19, acarretou numa grande quantidade de mortes decorrentes da doença. No Brasil, o primeiro caso de infecção foi no dia 26 de fevereiro de 2020 [UNA-SUS, 2020]. Nos meses que se sucederam, o país acumulou centenas de milhares de mortes e cerca de 10% da população foi infectada com o vírus até a presente data.

As medidas tomadas para conter o vírus impactaram a população em diversos níveis. Desde o individual, com o uso de máscaras, até o de países com lockdowns, fechamento de fronteiras e aeroportos. A resposta que melhor teria chance de contenção seria o desenvolvimento de vacinas e sua aplicação em massa, uma vez que historicamente as vacinas são boas saídas para erradicar doenças e combater epidemias [História das vacinas, 2021] e, segundo a epidemiologista Noronha [2021], a vacina é uma forma segura e eficaz de prevenir doenças e salvar vidas. Nesse sentido, cientistas de diferentes instituições se empenharam para desenvolver tais vacinas em tempo recorde e no final de 2020 as Campanhas de Vacinação foram iniciadas em alguns países como Estados Unidos, Israel e Inglaterra. No Brasil, a Campanha Nacional de Vacinação contra a COVID-19 se deu início em 17 de Janeiro de 2021 [G1, 2021].

Os registros de novos casos, número de mortes e de pessoas vacinadas gerou um enorme volume de dados e, consequentemente, houve um avanço na quantidade de pesquisas em torno do tema, incluindo pesquisas de Ciência de Dados para tentar extrair informações úteis e prever cenários na tentativa de contribuir para a redução do avanço da doença. Um exemplo é o trabalho de Chatterjee et al. [2020] que utiliza modelos Long Short Term Memory (LSTM) para prever novos casos e óbitos por COVID-19. O resultado mostra que modelos LSTM simples superam os modelos LSTM multicamadas.

O presente trabalho surgiu a partir de um projeto internacional para tratar questões da COVID-19 com Ciência de Dados. Esse projeto é intitulado como ICODA (International COVID-19 Data Alliance) e seus participantes utilizam a plataforma Workbench Aridhia

DRE[1] - específica para o domínio da saúde - para desenvolvê-lo. O projeto é uma iniciativa que apoia projetos de dados em saúde para responder de forma mais precisa à pandemia e a possíveis desafios no futuro, além disso, conta com o apoio da Fundação Bill & Melinda Gates, da Minderoo Foundation e do programa AI for Health, da Microsoft [Azevedo, 2021].

O objetivo do presente trabalho é a aplicação de metodologias de Ciência de Dados aos dados da Covid-19 a fim de analisar o impacto da vacinação no número de mortes pela doença. Para alcançar essa análise foi feita uma Regressão de Poisson buscando compreender se a variável de óbitos pode ser explicada pela variável da cobertura vacinal.

Para a realização deste trabalho, o mesmo foi organizado em 4 seções: Ciência de Dados, onde é explicado o conceito e o ciclo de vida da Ciência de Dados; Ferramentas de Apoio Computacional, mostrando alguns exemplos de Ambientes de Programação e Bancos de Dados Relacionais; Aplicação da Metodologia, onde é apresentado o passo a passo do ciclo de vida de Ciência de Dados aplicado ao projeto e, por fim, a Conclusão.

[1] https://www.aridhia.com/digital-research-environment/

# 2 CIÊNCIA DE DADOS

Para definir o que seria Ciência de Dados é importante trazer a diferença conceitual entre dados, informação e conhecimento. Entende-se por *dado* uma sequência de símbolos quantificados ou quantificáveis [Setzer, 2001]. Já a *informação* pode caracterizar-se como algo que é compreendido e tem sentido para alguém. Quanto ao conhecimento, Setzer o define como uma abstração interior de um indivíduo, algo que foi vivenciado. Theóphilo [1998] cita que o conhecimento costuma ser classificado de diferentes formas: popular, filosófico, teológico e científico. Para ele, o conhecimento científico, que aqui melhor se relaciona com a Ciência de Dados, diferencia-se dos demais não pelo objeto de estudo mas sim pela forma como é obtido.

A Ciência de Dados é uma área interdisciplinar que envolve Ciência da Computação, Estatística, Conhecimento específico do domínio, Engenharia de Dados e Desenvolvimento de Software [Shcherbakov et al. 2014]. Dessa combinação de disciplinas aplicada a grandes volumes de dados, sejam eles estruturados ou não, realiza-se a descoberta de padrões e informações relevantes o suficiente para tomadas de decisão e principalmente a construção de conhecimento. Para realizar tal Análise de Dados e, consequentemente, extrair informação, alguns passos são recomendados, tais como selecionar, preparar e transformar os dados, além de construir, avaliar e visualizar o modelo.

A Transformação Digital e o surgimento da Internet das Coisas aumentaram a utilização de dispositivos móveis e da internet, gerando assim um volume de dados em massa e, junto a essa realidade, o surgimento da Ciência de Dados foi consequência da necessidade de desenvolver técnicas capazes de trabalhar com esse grande conjunto de dados. Ezer e Whitaker [2019] definem o Cientista de Dados como desenvolvedor de novas técnicas de análise computacional ou estatística adaptáveis a diversos cenários, além de conseguir aplicar essas técnicas para responder a uma pergunta científica específica.

A Ciência de Dados vem ganhando forças e um dos motivos é o fato dela ter sido estruturada para resolver diversos tipos de problemas em muitos domínios diferentes. Por exemplo, utiliza-se modelagem estatística e/ou Machine Learning em cenários da saúde [de Souza et al. 2021], política [Awais et al. 2021], varejo [Zhao e Keikhosrokiani, 2022] , redes sociais [Amin et al. 2021], atendimento ao cliente [Golmohammadi et al. 2020], etc.

Olhando o caráter interdisciplinar desta ciência, é possível constatar o quanto pode ser relevante considerar a adição de suas técnicas no currículo de graduação de profissionais onde a metodologia científica é essencial na sua atuação. Por exemplo, no escopo do profissional

de engenharia de produção há como atividade o uso de métodos estatísticos[2], modelagem matemática e utilização de ferramentas computacionais, habilidades que podem ser facilmente voltadas para o objetivo final da Ciência de Dados. Ezer e Whitaker [2019], afirmam contudo, que ainda existem muitos profissionais, que não sabem como reformular suas questões de pesquisas como problemas de Ciência de Dados. Segundo Shcherbakov et al. [2014], nesse sentido ainda há duas questões em aberto: Encontrar a maneira adequada e eficiente de realizar pesquisas em Ciência de Dados e que tipo de abordagens podem ser adotadas para a construção de softwares que deem suporte a essas pesquisas?

## 2.1 Ciclo de vida da Ciência de Dados

O ciclo de vida da Ciência de Dados pode ser utilizado para auxiliar na tomada de decisão dos pesquisadores ou profissionais de uma Organização, além de ajudar a orientar as práticas de gerenciamento de dados.

Não há uma única definição acerca do ciclo de vida da Ciência de Dados, pode haver variação dependendo do conjunto de dados e do objetivo da análise. Stodden [2020] descreveu o ciclo de vida de Ciência de Dados nas seguintes etapas: projeto experimental; coleta de dados; exploração de dados e geração de hipóteses; limpeza, fusão e organização de dados; seleção de recursos e preparação de dados; estimação de modelo e inferência estatística; simulação e validação cruzada; visualização; publicação e preservação.

Shcherbakov et al. [2014] propuseram o ciclo de vida enxuto ("lean") da pesquisa em Ciência de Dados em dois níveis, externo ou macro e interno ou micro. Esse ciclo é apresentado na Figura 1.

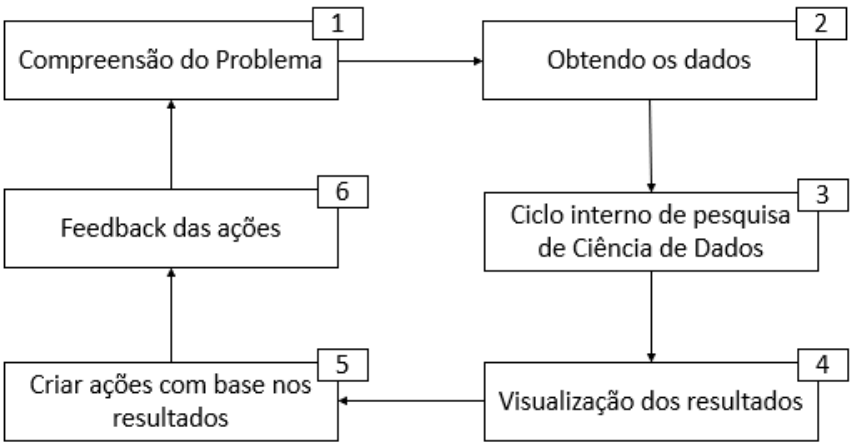


Figura 1 - Ciclo de Vida da Ciência de Dados

Fonte: Adaptado de Shcherbakov et al. (2014)

[2] http://www.puc-rio.br/ensinopesq/ccg/eng_producao.html#profissional

São seis etapas, as duas primeiras pertencem ao nível externo ou macro e o restante pertence ao nível interno ou micro. A primeira etapa consiste na compreensão do problema; a segunda é a obtenção dos dados e compreensão dos tipos de dados que são viáveis para a utilização da pesquisa; a terceira etapa é o ciclo interno que possui processos de análise de dados que vai desde o carregamento de dados no código até a obtenção dos resultados; a quarta etapa é a visualização dos resultados; a quinta é a criação de ações com base nos resultados e a sexta etapa consiste na obtenção do feedback da ação, podendo avaliar os resultados da análise com indicadores de performance.

### 2.1.1 Compreensão do problema

A compreensão do problema é fundamental para a análise, pois é aqui que são gerados os questionamentos e suas respectivas respostas, além de algumas decisões, como a definição do tipo de pesquisa mais apropriado para o problema. Alguns tipos de pesquisa são:

- Descritiva: Também é conhecida como Análise Exploratória e consiste na busca por padrões nos dados e por relação entre variáveis através de meios como por exemplo a análise estatística e visual;
- Preditiva: Consiste na busca de modelos de previsão considerando as variáveis e os dados históricos;
- Prescritiva: Consiste na busca de modelos que geram recomendações de ação.

### 2.1.2 Coleta de dados

Considerando que o problema já foi mapeado e o objetivo já está definido, na etapa de coleta de dados as partes envolvidas entendem e definem quais tipos de dados podem ser utilizados. Existem alguns tipos de dados, tais como:

- Dados Estruturados: São dados que possuem estruturas bem definidas. É o caso dos arquivos csv, excel e dos bancos de dados.
- Dados Semi-Estruturados: São dados com algumas características definidas, porém, com certa inconsistência. Um exemplo são os tweets.
- Dados Não Estruturados: Esses dados possuem uma formatação complexa que dificulta seu processamento. Alguns exemplos são os vídeos, sons e as imagens.

### 2.1.3 Ciclo interno de pesquisa

#### 2.1.3.1 Análise descritiva/exploratória

Esta etapa, tem por objetivo realizar visualizações de como se dispõem os dados e de forma resumida, compreender suas estatísticas básicas. Nela é realizada a descrição das variáveis (features) e é feito um mapeamento dos valores nulos. Além disso, são extraídas características como média, mediana e outliers. Outra atividade necessária aqui é a compreensão de como os dados se relacionam.

#### 2.1.3.2 Pré-processamento

Etapa que geralmente toma 80% das tarefas de um projeto em Ciência de Dados, o pré-processamento é essencial para que a próxima etapa (Modelagem dos Dados) seja efetuada satisfatoriamente. Nela são realizadas operações de preparação dos dados como a limpeza desses dados, podendo fazer a retirada de outliers, features que não serão úteis para o projeto e retirada de dados nulos ou o preenchimento dos mesmos. Nesta etapa também pode ser feita alterações na dimensionalidade, como por exemplo o agrupamento dos dados, além disso, é nesta fase que são realizadas mesclagens de dados de diversas fontes e geradas novas features.

#### 2.1.3.3 Modelagem

Esta fase caracteriza-se como aquela em que inicia-se a extração de padrões e informações, utilizando o poder computacional para modelar os dados de forma a classificá-los, agrupá-los ou realizar previsões a partir deles. Esta etapa pode ser dividida em selecionar um ou mais modelos, ajustar o modelo aos dados e avaliá-los.

#### 2.1.3.4 Interpretação dos resultados

Seu objetivo é entender se os resultados da etapa anterior devem ser utilizados ou não para a tomada de decisão. Para esta interpretação são utilizadas algumas métricas de avaliação dos modelos, como os critérios de informação, ou a medida a magnitude dos erros do modelo. Além disso, os resultados são comparados com benchmarks, ou seja, com padrões já conhecidos.

### 2.1.4 Visualização dos resultados

Nesta etapa são feitas as análises visuais dos resultados para apoiar a tomada de decisão. As visualizações podem ser feitas através de diferentes tipos de gráficos.

### 2.1.5 Tomada de decisão

Após a conclusão dos resultados, há duas possibilidades de caminhos a serem percorridos pelas partes envolvidas: buscar algumas ações para serem realizadas com base nos resultados ou melhorar a análise e repetir o ciclo.

### 2.1.6 Feedback

Nesta etapa observa-se, após o resultado da análise, o nível da qualidade do suporte à tomada de decisão, por exemplo através de indicadores de desempenho do negócio (key Performance indicators, ou KPIs).

## 2.2 Apoio Computacional

Dado um ciclo bem definido, onde as técnicas podem ser facilmente replicadas e os resultados minimamente aceitáveis, é preciso explorar e desenvolver ferramentas computacionais que deem suporte a essas pesquisas de forma a democratizar não apenas os dados mas também a reprodução do ambiente produtivo. Dessa forma, pessoas com baixo conhecimento computacional ou estatístico podem ter acesso às técnicas, métodos e algoritmos da Ciência de Dados.

Outro paradigma que deve ser observado quando em consideração a softwares é a rápida expansão do volume dos dados gerados com o passar dos anos [Martinez et al, 2021]. As soluções precisam suportar as atividades de armazenagem, tratamento e manipulação. Além do mais, é importante o desenvolvimento de modelos computacionais que sejam fáceis de replicar e gerar informação.

Em relação à coleta de dados, a grande utilização de sensores em devices como smartphones, televisores e automóveis, faz com que essa etapa do ciclo se torne mais rápida e eficiente do que já foi tempos atrás. Quanto à armazenagem desses dados, algumas tecnologias vêm se destacando como o Cloud Computing da AWS[3] (Amazon) ou o Azure[4] (Microsoft). Já para manipular e tratar esse volume, a alternativa utilizada é o processamento

[3] https://aws.amazon.com/pt/what-is-aws/
[4] https://azure.microsoft.com/pt-br/overview/what-is-azure/

paralelo como o do Hadoop[5], Apache Spark[6] e do Dask[7]. Algumas linguagens de programação se destacam quanto à modelagem como o Python [Python Software Foundation, 2021], SQL e R [R Core Team, 2021]. A aderência dessas linguagens se dá pela fácil utilização de suas bibliotecas como o Pandas [Reback et al., 2021], Scikit-Learn [Pedregosa et al., 2011], Tensorflow [Abadi et al, 2015], e ggplot [Wickham, 2009].

Após uma revisão das principais ferramentas utilizadas no mercado e indústria para as diversas etapas do ciclo de Ciência de Dados, Martinez et al. [2021] conclui que um framework para esse fim deve conter fundamentalmente os seguintes elementos: projeto, equipe e dados e gestão da informação.

A seguir, podemos verificar algumas categorias de tecnologias que ganharam aderência por conta do mundo de dados ou até mesmo surgiram para esse fim.

### 2.2.1 Ambientes de programação e de gestão de dados

Um ambiente de programação é a coleção de ferramentas utilizadas no desenvolvimento de software [Sebesta, 2018]. Tal ambiente pode consistir num editor de código, um compilador e alguma outra ferramenta de testes.

Para um cientista de dados, é imprescindível estar familiarizado com algum ambiente de programação ou até mesmo entender de lógica de programação, de modo a automatizar o trabalho a ser realizado, uma vez que a quantidade de bibliotecas e pacotes voltados para diversas etapas do pipeline de Ciência de Dados cresce no mercado.

Nas subseções seguintes apresentamos alguns componentes de ambientes de programação e de gestão de dados bastante utilizados no contexto de Ciência de Dados na atualidade.

#### 2.2.1.1 Sistemas Gerenciadores de Bancos de Dados (SGBDs) Relacionais

Desenvolvida na década de 70 dentro do programa System R da IBM, pelo cientista da computação Donald Chamberlin, a linguagem SQL é voltada para consultas estruturadas em bancos de dados relacionais [Todd, 1970]. É uma linguagem de fácil compreensão e muito utilizada na área de dados[8].

SQL é uma linguagem para definição e manipulação dos dados armazenados em Sistemas de Gerenciamento de Bancos de Dados. Os SGBDs possuem uma interface onde um

[5] https://hadoop.apache.org/
[6] https://spark.apache.org/
[7] https://docs.dask.org/en/latest/why.html
[8] https://www.dataquest.io/blog/why-sql-is-the-most-important-language-to-learn/

cliente, não necessariamente um programador, pode carregar, armazenar e modificar seus bancos de dados. Com o crescimento dos volumes de dados, a utilização desses softwares em projetos e organizações torna-se essencial.

- Oracle

Fundada em 1977, sua especialidade eram SGBDs e atualmente oferece uma série de infraestruturas voltadas para dados. Vale destacar que a Oracle é proprietária da linguagem de programação Java e do SGBD de código aberto MySQL, o mais utilizado no mundo, segundo informação em seu site institucional.

- My SQL

O gerenciador Oracle MySQL é um sistema de código aberto utilizado por empresas como Facebook, YouTube e LinkedIn, segundo informado em sua página institucional. O sucesso do SGBD vem da sua fácil integração com a linguagem PHP, utilizada na construção de sites. Atualmente oferece diversos serviços pagos e o MySQL Community como o único serviço gratuito.

- PostgreSQL

O sistema surgiu como um projeto da Universidade de Bekerley e abandonado, para em seguida se tornar um dos Bancos de Dados de código aberto mais utilizados pela comunidade de dados. A ferramenta se mantém por desenvolvedores voluntários e atualmente é utilizada por empresas [9] como a Apple, Cisco e Red Hat.

O programa, já na versão 14, oferece o pgAdmin, uma interface interativa, compreensível e que permite uma fácil configuração. Além de armazenar os dados com segurança e robustez[10].

- SQLServer

Tal SGBD surgiu de uma parceria entre as desenvolvedoras de software Sybase e Microsoft. Ao fim da parceria a manutenção da ferramenta ficou por conta da Microsoft.

[9] https://www.devmedia.com.br/guia/guia-de-postgresql/34328
[10] https://www.postgresql.org/about/

Atualmente o Microsoft SQL Server está na versão 19 e vem adequando cada vez mais serviços para as estruturas de internet. Com o surgimento do Microsoft Azure[11], serviço de cloud da empresa, o SQL Server serviu de base para o gerenciador de banco de dados da plataforma: o Azure SQL[12]. Contudo, para uso local, a empresa oferece um serviço pago[13] que pode interagir com outros bancos de dados além de ferramentas de big data.

#### 2.2.1.2 R

R é uma linguagem de programação voltada para computação estatística [R Core Team, 2021]. Além disso, um software livre disponível para os sistemas operacionais Windows, MacOS e Linux. Como um projeto do software livre GNU (GNU Não é Unix)[14], a linguagem R foi escrita por Robert Gentleman e Ross Ihaka e inspirada pela também linguagem de programação estatística S (de statistics) desenvolvida pela então empresa de telecomunicações Bell Laboratories.

No site do projeto ele é referenciado como um ambiente de forma a caracterizá-lo como um sistema que integra ferramentas, voltadas à dados e estatística, e que é planejado e coerente de forma a não ser como outras ferramentas muito específicas e inflexíveis. Dado tal ambiente, sua manutenção é realizada pela comunidade que utiliza o ambiente R. Esta comunidade também é responsável pela popularidade da linguagem.

- RStudio

Dentro do universo dos ambientes de programação estão as IDE's: Integrated Development Environment ou Ambientes de Desenvolvimento Integrado. São ambientes com interfaces gráficas e facilidades para desenvolvimento de aplicações. O RStudio é a IDE desenvolvida para a linguagem R [Allaire, 2012].

RStudio foi inspirado por profissionais da ciência, educação, governo e indústria integrando tecnologias de forma a facilitar os projetos voltados a dados. O software está disponível gratuitamente e na versão comercial para equipes, para desktop e online. Na sua última versão, a 1.4, além dos habituais recursos como editor de código com realce do texto,

[11] https://azure.microsoft.com/en-us/overview/
[12] https://azure.microsoft.com/pt-br/products/azure-sql/#product-overview
[13] https://www.microsoft.com/pt-br/evalcenter/evaluate-sql-server-2019?filetype=EXE
[14] http://www.gnu.org/

execução direta do código e ferramentas de plotagem, estão disponíveis também o R Markdown [15] e recursos para Python.

- Shiny

Dentre os pacotes mais utilizados na linguagem R está o Shiny. Desenvolvido pelo time do RStudio, o pacote permite a criação de aplicações interativas para web a partir dos dados. O Shiny se torna então uma opção para a etapa do pipeline de Ciência de Dados onde as análises são disponibilizadas e os modelos colocados em produção. Para a exposição desses aplicativos existe o shinyapps.io, um serviço de hospedagem para as aplicações criadas em Shiny.

- Tidyverse

O Tidyverse constitui uma coleção de pacotes de funções e métodos voltados para Ciência de Dados em R, dentre eles o ggplot2 [Wickham, 2012], uma poderosa ferramenta para criação de gráficos a partir de dataframes, e o manipulador de dados dplyr, entre outros.

#### 2.2.1.3 Python

A linguagem de programação criada em 1991 pelo programador e matemático holandês Guido van Rossum, é denominada como uma linguagem de alto-nível, ou seja, que tem por objetivo uma compreensão maior do programador que do computador. No site organizacional da linguagem, ela é definida como fácil de se aprender e amigável. Além disso, possui licença de código aberto e é aplicável a diversos tipos de projetos como jogos, web, processamento de textos, modelagem científica e outras.

Desde 2001 a licença e o desenvolvimento da linguagem é gerida pela Python Software Foundation[16], uma organização que realiza incentivos estudantis, administra os pacotes e suas documentações além de organizar a conferência anual da comunidade. Tal comunidade é responsável pela criação de novas bibliotecas que acabam por expandir tanto o uso da linguagem quanto as suas aplicações. As bibliotecas Python podem ser instaladas com a ferramenta pip[17] pelo prompt de comando.

---

[15] https://blog.rstudio.com/2020/09/30/rstudio-v1-4-preview-visual-markdown-editing/
[16] https://www.python.org/psf-landing/
[17] https://pypi.org/

- Projeto Jupyter

Originado no projeto IPython, o Projeto Jupyter[18] é uma iniciativa de construção de software aberto que tem por objetivo a utilização de diversas linguagens de programação para Ciência de Dados e computação científica. O nome  Jupyter é a junção das linguagens de programação Julia[19], Python e R.

Uma das ferramentas originadas no projeto são os arquivos Jupyter Notebook, ou simplesmente notebook. Um notebook é um ambiente composto por células que podem ser editores de código ou de texto, os chamados markdowns. Toda vez que um notebook é aberto ele inicia um kernel, uma máquina computacional capaz de rodar o código em Python. Os notebooks podem ser executados em browsers web ou em plataformas como o VS Code[20] ou o Google Colaboratory[21].

- Bibliotecas Python para Ciência de Dados

Segundo VanderPlas [2016], a utilidade da linguagem Python para projetos em Ciência de Dados provém do seu grande e ativo ecossistema de pacotes terceiros. Ele destaca os seguintes: Numpy, Pandas, Matplotlib e Scikit Learn.

O Numpy é o pacote básico para a manipulação dos objetos do tipo arrays. Tal pacote detém funções para a realização de operações matemáticas, principalmente com matrizes. Dessa forma, o Numpy acaba sendo um pacote utilizado em outras bibliotecas usadas para manipulação de dados.

Para a manipulação de dados estruturados o Pandas é o pacote em Python que se propõe a oferecer uma manipulação de dados relacionais ou rotulados de forma rápida e flexível. As estruturas utilizadas são as Séries e o Dataframes, capazes de armazenar dados provenientes em CSV, Excel e SQL.

Segundo Hunter [2007], o Matplotlib é um pacote de gráficos 2D, para geração de imagens com qualidade. Foi inspirado por uma biblioteca do Software Matlab e criada para visualizações interativas e customizáveis.

---

[18] https://jupyter.org/about
[19] https://julialang.org/
[20] https://code.visualstudio.com/
[21] https://colab.research.google.com/?utm_source=scs-index

- Scikit Learn

Scikit-Learn, segundo Pedregosa et al [2011], é um pacote Python que integra uma gama de algoritmos de Machine Learning, supervisionados e não-supervisionados, que podem ser facilmente aplicados e entendidos para ambientes científicos ou comerciais. Assim como outras bibliotecas e pacotes em Python, o pacote é de código aberto e seu desenvolvimento é gerenciado pela comunidade através do repositório Github.

### 2.2.3 Ambientes independentes de domínio para suporte a projetos de Ciência de Dados

Tendo conhecimento do ciclo de vida de Ciência de Dados e das linguagens de programação disponíveis no mercado que dão suporte a esse ciclo, é necessário saber as ferramentas que irão auxiliar na análise dos dados ou na implementação dos algoritmos. Nas subseções seguintes estão listadas e explicitadas algumas das ferramentas para Ciência de Dados.

#### 2.2.3.1 KNIME

A primeira versão da KNIME Analytics Platform[22] foi lançada em 2006 e desenvolvida por uma equipe de desenvolvimento de software especializada em aplicações farmacêuticas. O propósito era criar uma plataforma de código aberto para auxiliar na colaboração e pesquisa, dando suporte a grandes quantidades de dados, e hoje, em seu site oficial, promete executar projetos com bilhões de linhas. O KNIME é altamente escalável e abrange módulos de carregamento, transformação, análise e exploração visual de dados.

KNIME é escrito em Java e permite que fornecedores de software comercial possam adicionar wrappers - adicionar funcionalidades a outras classes - de tal modo que suas ferramentas sejam executadas a partir do KNIME. A plataforma possui uma comunidade de usuários que além de adicionar novas integrações, compartilha conhecimento com outros usuários através do Fórum KNIME.

[22] https://www.knime.com/

### 2.2.3.2 DATABRICKS Lakehouse Platform

Segundo o site oficial da empresa de dados e IA Databricks[23], ela foi criada em 2013 pelos mesmos desenvolvedores do Apache Spark. A plataforma Databricks é a primeira plataforma Lakehouse e diversas empresas a utilizam para fazer engenharia de dados de grande escala, Ciência de Dados, Machine Learning e Análise de Negócios. Databricks possui diversos parceiros globais, como Microsoft e Amazon.

Segundo Lorica et al. [2020], a arquitetura de gerenciamento de dados Lakehouse surgiu da necessidade de um sistema flexível e de alto desempenho, uma vez que os Data Warehouses não são recomendados para dados não estruturados, de alta variedade e volume, além disso, os Data Lakes, apesar de serem adequados para armazenar dados brutos com formatos variados, não suportam transações e dificultam leitura em lote e streaming. Assim, para contornar as limitações dos Data Lakes, foi criado o Lakehouse que possui diversas características, dentre as quais o suporte a transações para dar consistência na leitura e gravação dos dados, suporte de BI permitindo o uso dessas ferramentas e armazenamento com formatos abertos e padronizados de tal modo que bibliotecas como as do Python possam acessar os dados de forma eficiente.

Desse modo, o Databricks permite que cientistas, engenheiros e analistas de dados apliquem análises avançadas de Machine Learning e processamento de dados em escala, utilizem Deep Learning em interpretação de imagens, tradução automática e processamento de linguagem natural, além de tornar o armazenamento de dados rápido, simples e escalável.

### 2.2.3.3 HADOOP

Hadoop foi lançado em 2008 pelo Yahoo, é um software de código aberto para computação distribuída confiável e escalável que permite o processamento distribuído de grandes conjuntos de dados em clusters de hardwares comuns e possui alta capacidade de armazenamento massivo para qualquer tipo de dado.

Além do grande poder computacional para processar Big Data de forma rápida, o Hadoop foi projetado para detectar e tratar falhas, armazenar dados sem precisar pré-processar e disponibilizar a estrutura de código aberto de forma gratuita. O Hadoop possui alguns

[23] https://databricks.com/

módulos, tais como Hadoop Distributed File System (HDFS) que fornece acesso de alto rendimento aos dados, Hadoop YARN que agenda tarefas e gerencia recursos de cluster, e Hadoop MapReduce para processamento paralelo de grandes conjuntos de dados.

Apesar dos benefícios mencionados, o Hadoop possui alguns desafios, uma vez que o MapReduce não é eficiente para análises avançadas baseadas em algoritmos iterativos. Outro ponto é que o Hadoop não dispõe de recursos intuitivos e completos para gerenciamento de dados, governança e metadados.

#### 2.2.3.4 Rapid Miner

O RapidMiner[24] foi desenvolvido em 2001 e é uma ferramenta de mineração de dados baseada em Java que fornece um ambiente integrado para mineração de texto, mineração de dados, Machine Learning, análise preditiva e de negócios, sendo usado principalmente para negócios e aplicações industriais [Chauhan  e Gautam, 2015].

De acordo com seu site, o RapidMiner promete ter a profundidade necessária para cientistas de dados e a simplicidade para todos os outros, uma vez que possui suporte para qualquer biblioteca de Machine Learning, integração com Python e R e análises avançadas, ao mesmo tempo que tem algumas simplificações, como modelos de casos de uso predefinido e tutoriais abrangentes.

#### 2.2.3.5 Microsoft Excel

O Microsoft Excel foi desenvolvido em 1987 e é, atualmente, um dos softwares mais populares no domínio de dados. De acordo com Moeschlin [2018], o Excel é o melhor editor de dados bidimensionais - com exceção do Big Data -  por sua facilidade de utilização e clareza do design. O Excel possui o Analysis ToolPak que auxilia nas análises estatísticas complexas, além disso,  o Excel possui integração com ambientes de programação como o Python e  tem a possibilidade de executar consultas SQL em tabelas. No entanto, não dá suporte às técnicas de modelagem e extração de conhecimento a partir dos dados.

[24] https://rapidminer.com/

### 2.2.4 Ambientes para suporte a projetos de Ciência de Dados em domínios específicos

#### 2.2.4.1 A Plataforma Aridhia

O Workbench Aridhia DRE (Digital Research Environment)[25] é um ambiente virtual de pesquisa projetado especialmente para desenvolvimento de projetos de Ciência de Dados no domínio da Saúde. A plataforma hospeda os chamados workspaces, que são espaços onde equipes alocadas a projetos podem organizar seus artefatos contando com ferramentas específicas. O Aridhia Workspace pode ser acessado por um navegador da web.

Tecnicamente falando, especificamente para o projeto ICODA\EFFECT-Brazil, este workspace oferece até 5 TB de armazenamento em seu servidor, para cada projeto. Todos os *datasets* são nativamente armazenados em PostgreSQL e podem ser consultados via SQL. Além disso, é possível utilizar máquinas virtuais, Python e R.

Em seu vídeo institucional, a empresa se propõe a oferecer suporte a todo o ciclo de vida de Ciência de Dados. A plataforma disponibiliza recursos e ferramentas como o Analyse Data que permite a geração automática de gráficos a partir de datasets, o Jupyter Notebook que permite criar códigos em algumas linguagens como Python, além de uma interface para criação de consultas SQL e uma Máquina Virtual que possui acesso a todas as pastas do projeto e essas ferramentas citadas, dentre outros recursos.

Como parte desse trabalho, diversos recursos da plataforma Aridhia foram avaliados, no escopo do projeto ICODA\EFFECT-Brazil, para analisar e extrair conhecimento a respeito do processo de vacinação contra a COVID-19 nos municípios do Brasil.

Algumas limitações ou erros de execução foram observados, listados a seguir:

- Ao criar uma consulta SQL a partir de uma custom view, a funcionalidade de editar/alterar o código apresentava-se inoperante, fato que foi repostado ao suporte da ferramenta.
- A função Analyze Data é restrita para consultas com até 10 mil linhas e 100 colunas. Ao analisarmos a mensagem de erro “Read Timeout with # <TCPSocket:(closed)>”, supomos que este erro é gerado pela lentidão do servidor e consequência da combinação do grande conjunto de dados e complexidade da consulta SQL. Tal limitação restringiu o seu uso no projeto,

[25] https://www.aridhia.com/

uma vez que as tabelas a serem analisadas continham uma quantidade de dados na ordem de milhões de linhas.

- Ao utilizar o Jupyter Notebook dentro da plataforma Aridhia, a análise do conjunto de dados de vacinação não foi bem sucedida. Apenas foi possível ler o dataset e contabilizar as linhas nulas por coluna, porém, ao tentar dividir esse valor pelo número total de linhas para obter o percentual de nulos, aparece a mensagem sinalizando que o kernel está sendo reiniciado, o que significa que a memória RAM já foi consumida. Ou seja, não é possível analisar toda a base de vacinação do Brasil através do Jupyter dentro do Aridhia.

#### 2.2.4.2 SageMaker AWS

O SageMaker AWS[26] é o primeiro ambiente de desenvolvimento integrado (IDE) para Machine Learning, foi lançado em 2017 com o objetivo de auxiliar seus usuários na preparação, criação, treinamento e implementação de modelos de ML de forma rápida e com alta qualidade. O SageMaker possui ferramentas específicas para cada etapa de desenvolvimento de Machine Learning, como o SageMaker Experiments que captura, organiza e compara cada passo.

#### 2.2.4.3 Google Cloud Platform

É um ambiente[27] para armazenamento e manipulação de dados oferecido pelo Google com acesso aos demais serviços da empresa como o YouTube e o seu buscador. A plataforma oferece também ferramentas de machine learning e de Business Intelligence. Além disso, é altamente voltado para o suporte à tomada de decisão de empresas que queiram adotar a cultura de dados.

### 2.2.5 Sistemas de gerência de workflows científicos

Os Sistemas de Gerência de Workflows Científicos (SGWfC) auxiliam no desenvolvimento do processo de workflow voltado para as aplicações científicas de qualquer área de pesquisa [Mattos et al. 2008]. Sendo assim, a partir dos workflows, essas aplicações científicas podem ser representadas em forma de conjunto de tarefas, porém, podem necessitar de um alto apoio computacional, uma vez que os workflows costumam ser

[26] https://aws.amazon.com/pt/sagemaker/
[27] https://cloud.google.com/why-google-cloud

utilizados com grandes conjuntos de dados. Segundo Silva (2007), um Workflow Científico possui algumas características, dentre as quais algumas são o foco em processos centrados em dados e a possibilidade de modificação do workflow durante sua execução.

Os SGWfC auxiliam também no monitoramento e visualização do processo de workflow, ou seja, permitem gerenciar o ciclo de vida de um Workflow Científico. Esse ciclo de vida pode ser representado pela figura 2 segundo a adaptação do Ludäscher et al. [2009] retirada de Vieira e Nascimento [2019].

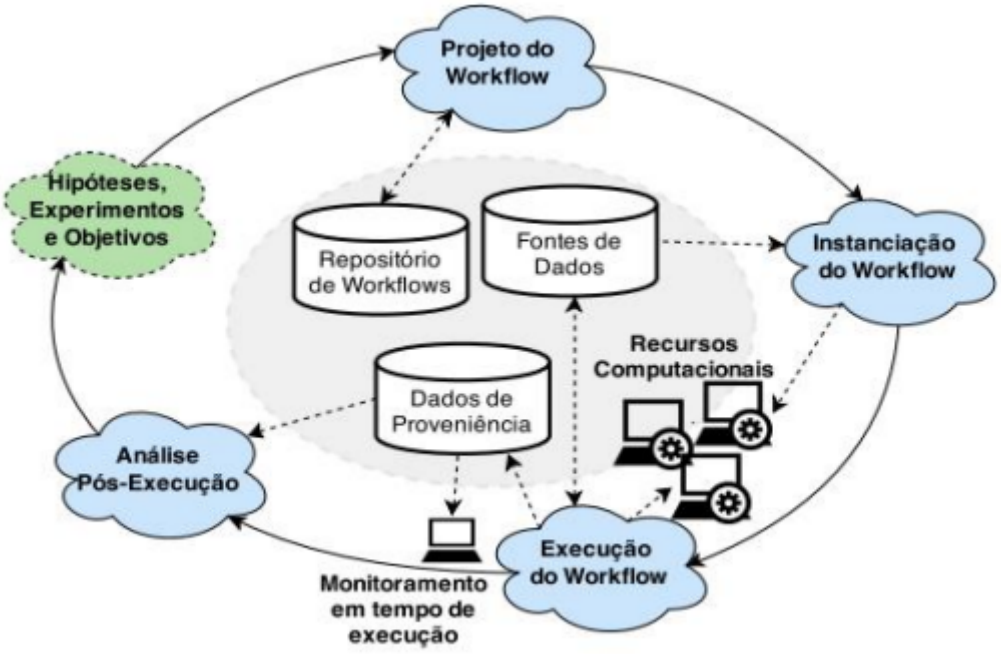


Figura 2: Ciclo de vida de um Workflow Científico
Fonte: Ludäscher et al. (2009)

- Projeto do Workflow: nesta etapa, com base no objetivo do projeto, é definido e construído um modelo para o workflow. Como um projeto de workflow pode ser baseado em outro, é recomendado que esse projeto seja salvo e até compartilhado com outros pesquisadores.
- Instanciação do Workflow: aqui acontece a preparação necessária para a execução particular do workflow, como o input dos dados e definição dos parâmetros do modelo.
- Execução do Workflow: nesta fase os dados de entrada são consumidos e novos dados são gerados. Além disso, ocorre o monitoramento do tempo de execução e dos possíveis problemas que venham surgir.

- Análise Pós-Execução: nesta etapa os cientistas analisam os dados resultantes da execução do workflow. Dependendo do resultado, as hipóteses e os objetivos definidos na fase de Projeto do Workflow podem ser revistos e o ciclo de vida reiniciado.

De modo a exemplificar os conceitos já descritos, a seguir serão apresentados alguns sistemas de gerenciamento de Workflows Científicos existentes na literatura.

### 2.2.5.1 myExperiment

O myExperiment[28] é um ambiente online com a finalidade de colaboração. Nele os usuários têm a possibilidade de compartilhar seus Workflows, acessar os de outros colaboradores, criar comunidades ou acessar grupos já existentes. Dentro do myExperiment há diversos tipos de Workflows, dentre os quais alguns são o Taverna, Galaxy, Rapid Miner, Bio Extract e Kepler.

A plataforma do myExperiment foi divulgada em 2007 e surgiu a partir do consórcio myGrid formado por colaboradores de diversas Universidades  e parceiros industriais. Em 2015 esse consórcio formou o e-Science Lab que atualmente tem financiamento até 2023 e suportam diversos softwares que foram desenvolvidos, incluindo o myExperiment.

### 2.2.5.2 Weka Experimenter e Knowledge Flow

Waikato Environment for Knowledge Analysis (WEKA), teve seu projeto iniciado em 1992 devido a necessidade de ter um workbench unificado para acesso rápido às técnicas de Machine Learning [Witten et al. 2009]. O WEKA é um software de código aberto e, segundo R Development Core Team [2008], é reconhecido como um sistema de referência em mineração de dados e aprendizado de máquina.

O WEKA oferece diversas interfaces gráficas, como por exemplo o Knowledge Flow e Experimenter. O Knowledge Flow consegue trabalhar com grandes conjuntos de dados e, além de possuir um treinamento baseado em lote, tem um modelo de fluxo de dados como exemplificado na figura 3. O Weka Experimenter facilita a comparação dos experimentos de algoritmos se baseando em critérios de avaliação do próprio WEKA.

[28] https://esciencelab.org.uk/products/myexperiment/

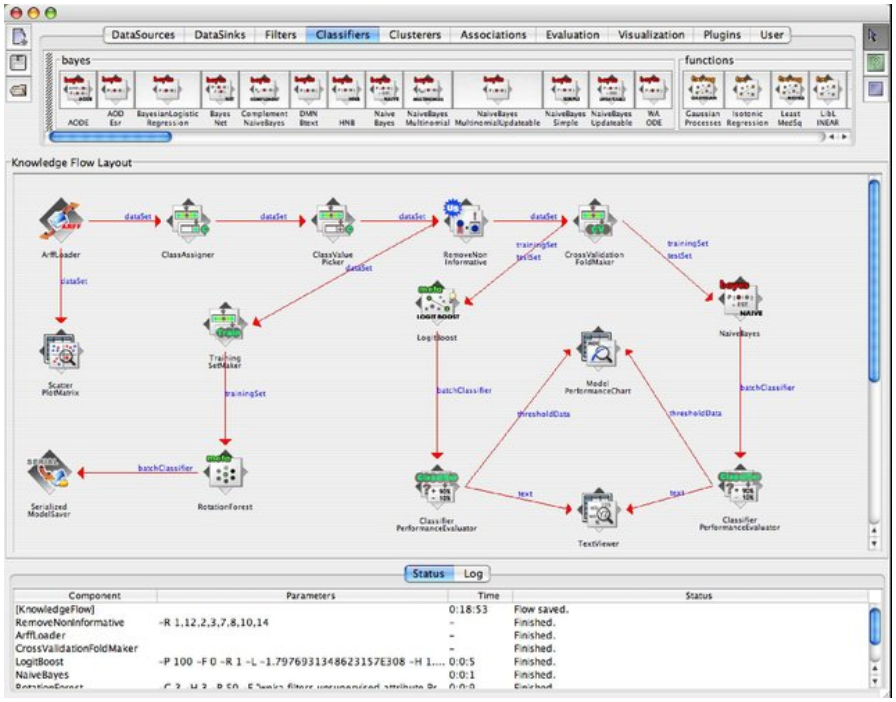


Figura 3: WEKA Knowledge Flow
Fonte: Witten et al. (2009)

### 2.2.5.3 Pegasus

Atualmente a ciência requer uma simulação em grande escala e análise de dados para estudar sistemas complexos, para isso geralmente são necessários sistemas de gerenciamento de Workflows escaláveis que sejam capazes de coordenar e automatizar a movimentação de dados e a execução de tarefas de maneira confiável e eficiente em recursos computacionais distribuídos [Deelman et al. 2014]. Nesse contexto foi desenvolvido pela primeira vez o Sistema de Gerenciamento de Workflow Pegasus em 2001[29], com o intuito de unir o domínio científico e a infraestrutura cibernética disponível.

O Pegasus executa aplicativos baseados em Workflows em diversos ambientes diferentes, como desktop e nuvens, além de localizar os dados de entrada e recursos computacionais necessários para executar o Workflow. No Pegasus os Workflows são representados como gráficos acíclicos direcionados onde os nós são tarefas computacionais individuais e as bordas representam dados e dependências de controle entre as tarefas. O Pegasus permite construir um Workflow abstrato, ou seja, sem mencionar informações detalhadas como as de recursos e dados.

[29] https://pegasus.isi.edu/

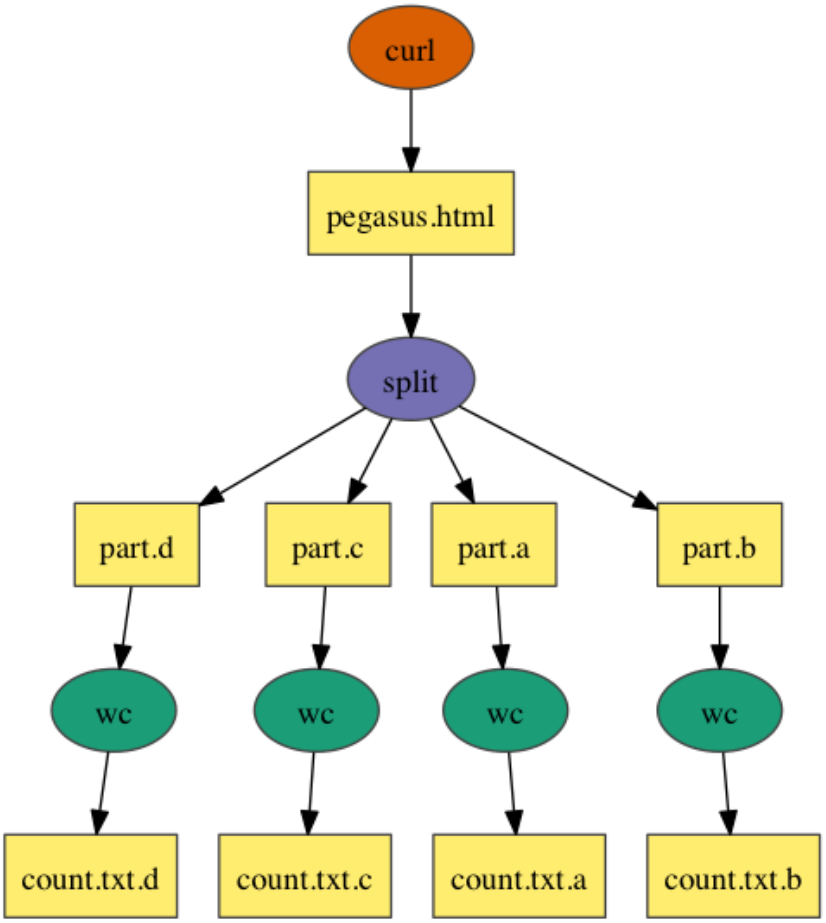


Figura 4: Exemplo de workflow - Pegasus

Fonte: https://pegasus.isi.edu/documentation/examples/

### 2.2.5.4 Kepler

O Kepler[30] foi criado em 2002 e combina o design de workflow de alto nível com execução e interação em tempo de execução e acesso de dados locais e remotos [Altintas et al. 2004], além de ser um ambiente eficaz para integrar componentes de softwares distintos, como mesclar scripts “R” com código “C”. O Kepler auxilia os usuários no compartilhamento e reutilização de dados e Workflows, e foi baseado no sistema Ptolomeu II[31] que tem ênfase na interação de múltiplos componentes.

[30] https://kepler-project.org/
[31] http://ptolemy.eecs.berkeley.edu/ptolemyII/

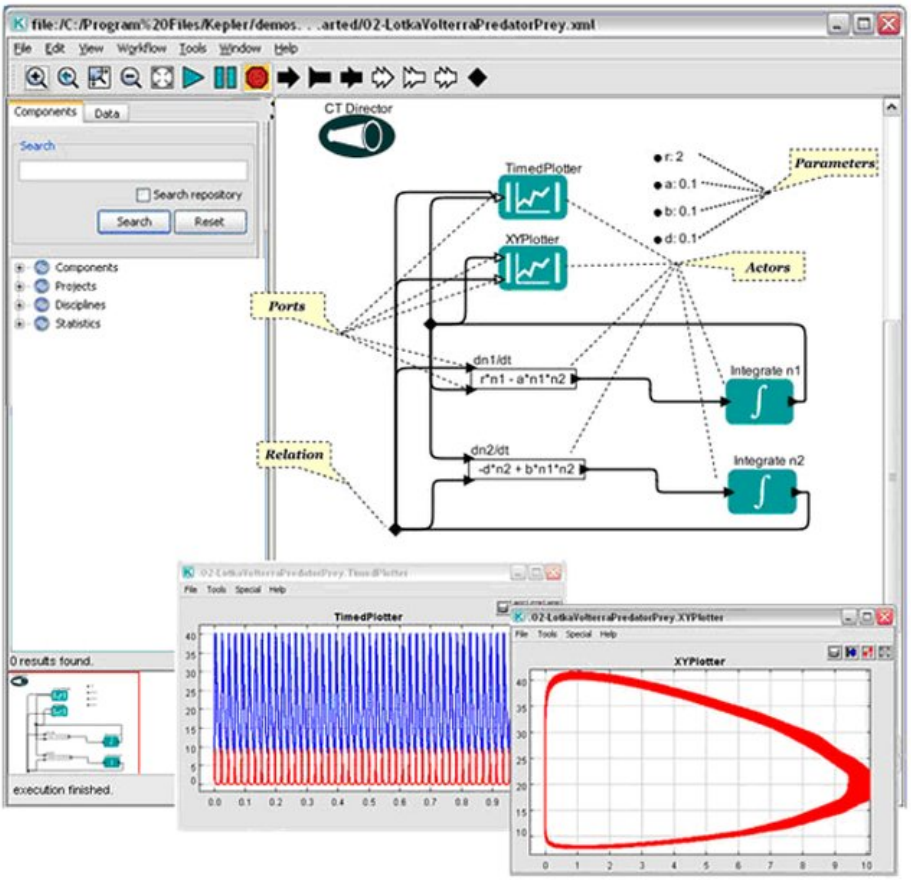


Figure 5: exemplo de workflow modelado com o Kepler

Fonte: Braghetto e Cordeiro (2014)

### 2.2.5.5 SAMbA-RaP

SAMbA-RaP (Spark provenAnce MAnagement with Reports and Presentation) estende o Apache Spark[32] para gerenciamento de tipos de dados de proveniência. Os dados produzidos do compartimento RDD (Resilient Distributed Datasets) são armazenados pelo SAMbA-RaP de forma estruturada e de tal modo que os dados de proveniência coletados possam ser consultados em tempo de execução ou após a execução do Workflow [Guedes et al. 2020]. Além disso, o SAMbA-RaP permite o rastreamento e armazenamento gradual dos dados no servidor de dados de proveniência.

[32] https://spark.apache.org/docs/latest/cluster-overview.html

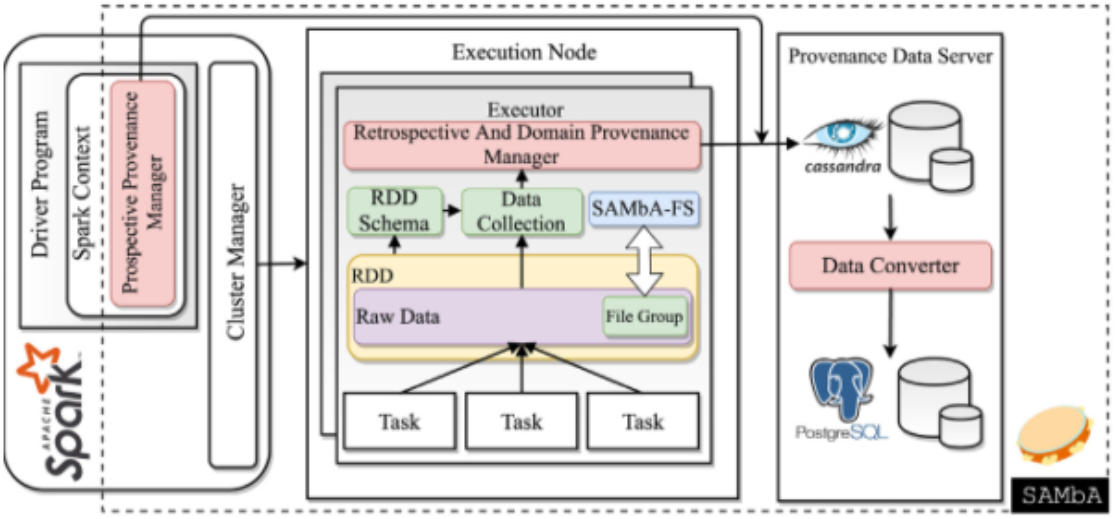


Figura 6: Arquitetura SAMbA-RaP combinada com Apache Spark

Fonte: Guedes et al. (2020)

# 3 CIÊNCIA DE DADOS APLICADA AO PROJETO ICODA\EFFECT - Brazil: UMA ANÁLISE DO IMPACTO DA VACINAÇÃO NA MORTALIDADE POR COVID-19 EM MUNICÍPIOS BRASILEIROS USANDO A LINGUAGEM DE PROGRAMAÇÃO PYTHON E O SGBD POSTGRESQL

Com o avanço da COVID-19 no Brasil e no mundo, massivos conjuntos de dados vêm sendo disponibilizados e, em paralelo, há uma crescente profusão de ambientes de apoio computacional para iniciativas de Ciência de Dados, conforme apresentado na Seção 2. Consequentemente, as pesquisas e artigos em torno do tema têm sido bastante frequentes, um exemplo de pesquisa é o trabalho do de Souza et al. [2021] que utilizou técnicas de clusterização sobre séries temporais de novos casos de COVID-19 nos estados do Brasil e foi concluído que existem seis grupos de estados com evoluções semelhantes da doença. Dado o início e progresso da vacinação, começaram também os estudos de modo a compreender uma série de questões sobre a vacina. Neste sentido, a presente seção tem como objetivo detalhar as etapas da aplicação do ciclo de vida de Ciência de Dados no escopo do projeto "ICODA\EFFECT-Brazil", com o objetivo de analisar o impacto da vacinação no Brasil na mortalidade por COVID-19 a nível municipal.

As Seções a seguir relatam a execução de cada etapa do ciclo de vida de Ciência de Dados, descrito na Seção 2.1, no escopo proposto.

## 3.1. Compreensão do problema

No seu surgimento, a COVID-19 trouxe uma série de dúvidas em relação ao desenvolvimento da doença e seus efeitos, tanto para cada indivíduo como para a Sociedade como um todo. Tal cenário trouxe uma série de incertezas em relação ao enfrentamento [Barreto-Filho, 2020]. Diversas medidas foram tomadas com a intenção de conter o avanço do vírus e o efeito de cada medida pôde ser percebido com o decorrer do tempo. Contudo, a maior expectativa foi depositada nas campanhas de vacinação em massa.

Historicamente, o uso de vacinas tem se mostrado como uma boa opção para a erradicação de doenças [d'Avila, 2019]. Para responder tal demanda, alguns laboratórios farmacêuticos, unidos a cientistas de universidades e centros de pesquisas no mundo inteiro, engajaram-se para produzir um imunizante eficaz contra a COVID-19. No dia 17 de Janeiro de 2021, uma enfermeira recebeu, na cidade de São Paulo, a primeira dose do imunizante CoranaVac, desenvolvido pelo laboratório chinês Sinovac em parceria com o Instituto

Butantan [Istoé, 2021]. Outros imunizantes chegaram ao país e foram aprovados pela Anvisa, agência que regula os medicamentos no Brasil. Foram eles: os do laboratório Astrazeneca [World Health Organization, 2021], desenvolvido em parceria com a Universidade de Oxford e a Fundação Oswaldo Cruz; da farmacêutica Pfizer[33] e o imunizante Janssen [Ninomiya, 2021], da fabricante Johnson & Johnson. Com exceção da vacina Janssen, de uma dose, para os demais imunizantes são necessárias duas doses para que seja atingido seu potencial. As vacinas desses laboratórios foram distribuídas entre os Estados e municípios, de forma que cada prefeitura organizasse sua aplicação levando em conta critérios como: comorbidades, grupos de profissionais e faixa etária. A co-existência de diferentes fabricantes e tipos de vacinas, com número de doses variável e diferentes estratégias de vacinação adotadas para cada Estado (e eventualmente até para cada município do mesmo Estado), diferindo-se quanto aos critérios de prioridade e à velocidade de vacinação, tornou a gestão e o monitoramento dos efeitos da vacinação (com respeito à eficácia, efetividade e impacto) uma questão muito complexa.

Aqui é importante trazer uma explicação a respeito dos conceitos de eficácia, efetividade e impacto da vacinação. Segundo Ortiz e Neuzil [2021], a eficácia de uma vacina é determinada por ensaios clínicos randomizados, realizados em uma fase de testes em laboratório; a efetividade é estimada a partir de estudos observacionais após o início da vacinação, ainda com relação aos efeitos da vacina em nível individual. Já o impacto da vacinação é uma medida utilizada para estimar a redução de doenças em uma comunidade.

Neste trabalho, a proposta é analisar o impacto da Campanha de Vacinação contra a COVID-19 no número de óbitos registrados nos municípios brasileiros a partir da aplicação do Ciclo de Vida de Ciência de Dados, de forma a aproveitar o poder dos dados e colaborar na iniciativa global ICODA, visando apresentar informações relevantes para órgãos públicos e instituições voltadas ao enfrentamento da COVID-19.

## 3.2 Coleta de Dados

Para endereçar a questão de pesquisa definida na seção anterior, foram levantadas informações[34] sobre bases de dados públicas a respeito tanto da evolução da COVID-19 em cada município brasileiro (número diário de casos, óbitos e vacinados, hospitalizações),

[33] https://www.pfizer.com.br/pfizer-no-combate-ao-coronavirus
[34] Análise Descritiva - Bases ICODA

quanto de informações sociodemográficas dos municípios. Neste sentido as fontes de dados escolhidas foram: SIPNI, Brasil.IO - casos_full e IBGE, que são detalhadas a seguir.

- SIPNI[35] - Sistema de Informações do Programa Nacional de Imunizações

O SIPNI consiste em um conjunto de sistemas que abrangem informações dos programas de imunização realizados no país. Este sistema é responsável por disponibilizar os dados da Campanha Nacional de Vacinação contra a COVID-19 desde o seu início.

Os dados do SIPNI são disponibilizados em arquivos CSV, com os dados separados por cada Estado ou completo (união de todos os Estados). Tais dados descrevem cada dose aplicada nos indivíduos residentes em cada UF (ou em todo o Brasil), desde 17/01/2021. Os dados são anonimizados e apresentam informações básicas sobre o paciente como idade e cep, sobre o estabelecimento de saúde onde a dose foi aplicada e sobre a vacina aplicada. A atualização diária desta fonte de dados faz com que o volume dos dados cresça rapidamente e impossibilite a utilização de algumas das ferramentas e ambientes de suporte à Ciência de Dados descritos na Seção 2.2, que comumente são utilizadas em projetos descritos na literatura da área.

Até o dia 30/10/2021, haviam cerca de 266 milhões de registros de doses de vacinas de COVID-19 cadastradas no SIPNI. Dessa forma, para melhor manipulação destes dados, este trabalho se restringiu à avaliação dos 3 Estados com maior e os 3 com menor cobertura vacinal, até o dia 30/10/2021[36], além do Estado do Rio de Janeiro, por ser o local onde os integrantes desse trabalho residem. A maior cobertura vacinal pode trazer maior indicação do impacto da vacina na redução dos óbitos, enquanto que os Estados de menor cobertura servem como comparativo. Os Estados que atenderam aos critérios mencionados foram São Paulo, Santa Catarina e Mato Grosso do Sul com as maiores coberturas vacinais; Amapá, Roraima e Pará com menor cobertura vacinal.

- Brasil.IO - caso_full[37]

O Brasil.IO é um portal web mantido por um grupo de desenvolvedores cuja missão é "tornar acessíveis os dados brasileiros de interesse público", prezando pela transparência e colaboração e com a intenção de consolidar e democratizar os dados, disponibilizando-os gratuitamente para download. Através deste portal, pode-se ter acesso a um repositório de

[35] http://sipni.datasus.gov.br/si-pni-web/faces/apresentacaoSite.jsf
[36] https://especiais.g1.globo.com/bemestar/vacina/2021/mapa-brasil-vacina-covid/
[37] https://brasil.io/dataset/covid19/caso_full/

dados abertos do Brasil, composto por diversos datasets sobre os mais variados temas, dentre eles a COVID-19. Para esse trabalho, o dataset escolhido foi o “caso_full”, o qual é uma consolidação dos dados de morte e casos de COVID-19 no Brasil, agrupada por município e semana epidemiológica. Este dataset é estruturado como uma tabela, cujos campos podem ser visualizados na tabela do anexo.

- IBGE

O Instituto Brasileiro de Geografia e Estatística é um órgão federal responsável por realizar pesquisas e consolidar dados que dizem respeito à economia, sociedade e geociências do país. O instituto disponibiliza esses dados em diversas tabelas separadas por temas, estados, municípios e períodos. Para este trabalho, foi utilizada a informação de população por município em uma tabela com um compilado dos dados sociodemográficos. Tal tabela é um produto de uma das frentes de trabalho do Projeto ICODA e será aqui referenciada como “Dados_Sociodemográficos”.

Para a manipulação dessas bases foram utilizadas as plataformas Excel, o SGBD PostgreSQL e algumas bibliotecas Python específicas para a manipulação de dados como Pandas, DASK e Psycopg2.

## 3.3 Ciclo interno de pesquisa

### 3.3.1 Análise descritiva/exploratória

Após coletar os dados mencionados na seção anterior, foi feita uma análise descritiva de cada fonte de dados, a fim de obter informações estatísticas e compreender esses dados para poder tratá-los de forma adequada.

- SIPNI - Sistema de Informações do Programa Nacional de Imunizações

O grande volume de dados da base do SIPNI dificulta a manipulação dos dados até mesmo em bibliotecas voltadas para ambientes Big Data, como o Dask. A dificuldade persistiu mesmo após a seleção dos 7 Estados já mencionados (RJ, SP, SC, MS, PA, AP e RR), pois somente a base correspondente ao Estado de São Paulo possuía, no momento em que obtivemos, 32GB e cerca de 72 milhões de linhas. Para tentar contornar essa dificuldade, foi feito um tratamento de filtros e agrupamentos no PostgreSQL que será detalhado na próxima seção. Esse tratamento permitiu uma redução considerável na dimensão dos dados, levando São Paulo, por exemplo, a 1,9 milhões de linhas.

Originalmente, a base SIPNI possui 32 colunas que são apresentadas e descritas na Tabela 1 do anexo. Após alguns tratamentos realizados na base de acordo com nossa necessidade, obtivemos uma base intermediária do SIPNI com 7 Estados e, a partir dela, foi gerada uma análise descritiva desses dados.

A base ficou com 4 milhões de linhas e 11 colunas que correspondem ao código e nome do município do estabelecimento, UF do estabelecimento, semana epidemiológica, sexo biológico e raça do paciente, categoria e grupo de atendimento do vacinado, descrição da dose, nome da vacina e quantidade de vacinados.

Após importar os dados do PostgreSQL com a biblioteca Psycopg, foi identificado que somente as colunas *paciente_enumsexobiologico, paciente_racacor_valor e vacina_categoria_nome* possuem dados nulos, e ainda assim com pouca representatividade, cerca de 0,008% para as colunas de sexo e raça, e 0,62% para a coluna de categoria da vacina.

As colunas de código e nome do Município possuem 1.270 valores únicos, representando a quantidade de Municípios da base; a coluna de *estabelecimento_uf* possui as 7 Unidades Federativas escolhidas para análise no presente trabalho; a coluna de semana epidemiológica contém valores que variam de 202103 a 202139, representando a semana epidemiológica de 2021 para o período selecionado; a coluna de sexo possui, além dos sexos feminino e masculino, 100 linhas de Não Informado; a coluna de raça contém as raças branca, parda, amarela, preta e indígena, além de uma classe de sem informação correspondendo a cerca de 824 mil linhas; a coluna de *vacina_categoria_nome* representa o motivo pelo qual o paciente foi vacinado, se foi vacinado pela faixa etária, por comorbidades ou por ser profissional da saúde, além de diversas outras categorias; a coluna de *vacina_grupoatendimento_nome* é uma especialização da anterior, diz, por exemplo, qual a faixa etária do vacinado ou a comorbidade específica; a coluna de *vacina_descricao_dose* diz qual a dose correspondente da vacina (única, primeira, segunda ou dose de reforço); a coluna de *vacina_nome* representa qual foi a vacina aplicada (astrazeneca, coronavac, pfizer ou janssen); já a coluna de quantidade de vacinados é um campo que foi calculado no PostgreSQL e corresponde a quantidade de vacinas aplicadas levando em consideração o agrupamento realizado.

A única coluna com dados numéricos é a coluna de quantidade de vacinados, então para uma análise exploratória é possível extrair informação desse atributo levando em consideração as outras colunas categóricas. Na figura 7 é possível observar que a maior parte da população foi vacinada pela faixa etária e a segunda maior parcela foi vacinada devido a

alguma comorbidade. Já na figura 8 pode-se notar que a vacina mais distribuída nos Estados analisados foi a astrazeneca.

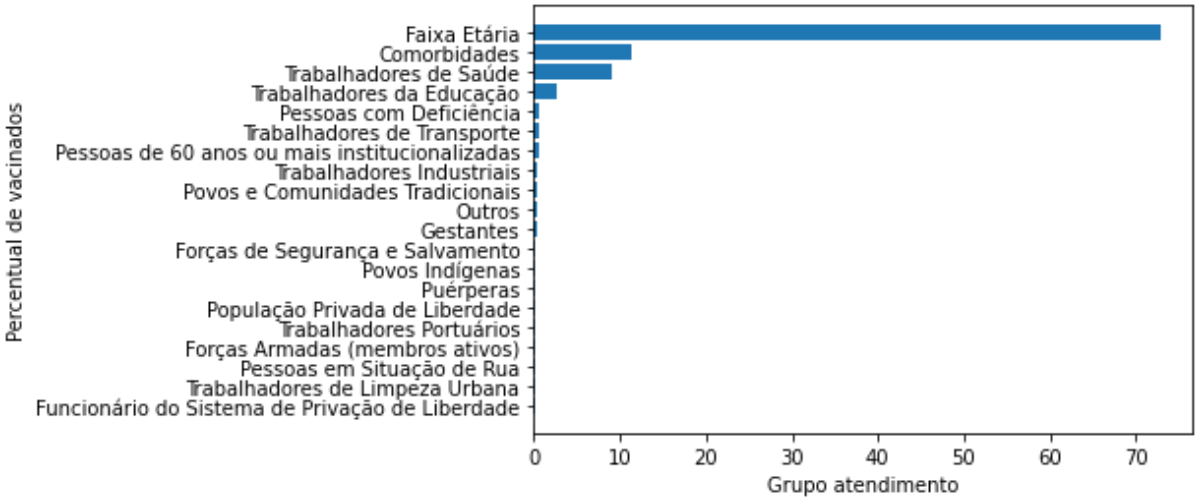


Figura 7: Percentual de vacinas aplicadas por grupo de atendimento

Fonte: Autores

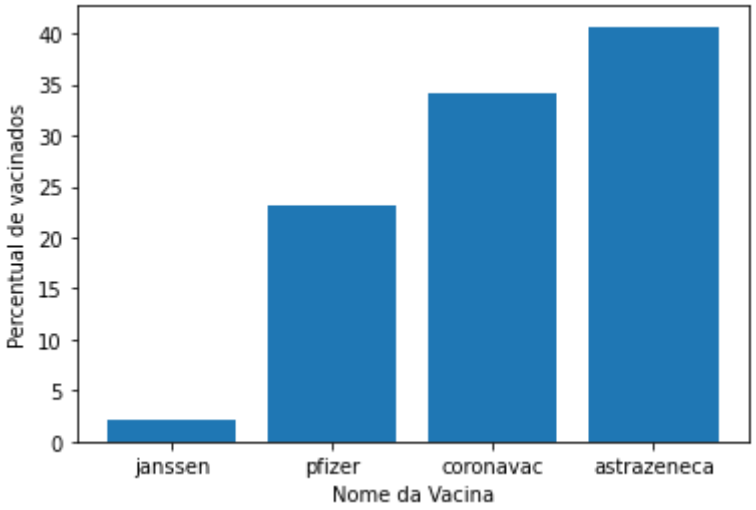


Figura 8: Percentual de vacinados por tipo de imunizante.

Fonte: Autores

- Brasil.IO - caso_full

A base do Brasil.IO possui cerca de 3 milhões de linhas e 18 colunas, seus dados representam os óbitos e número de casos de Covid-19 referente a todos os Municípios do Brasil. Os detalhes acerca do que cada coluna desse conjunto de dados representa pode ser encontrado na Figura 9 do anexo.

Na figura 9 é possível observar que cinco colunas possuem dados faltantes, sendo estes pouco representativos, uma vez que a coluna com o maior percentual possui um pouco mais de 0,8% de dados nulos. Esse conjunto de dados possui informações a partir da semana 9 de 2020, totalizando, até o momento da coleta desse dado, 89 semanas e 620 dias.

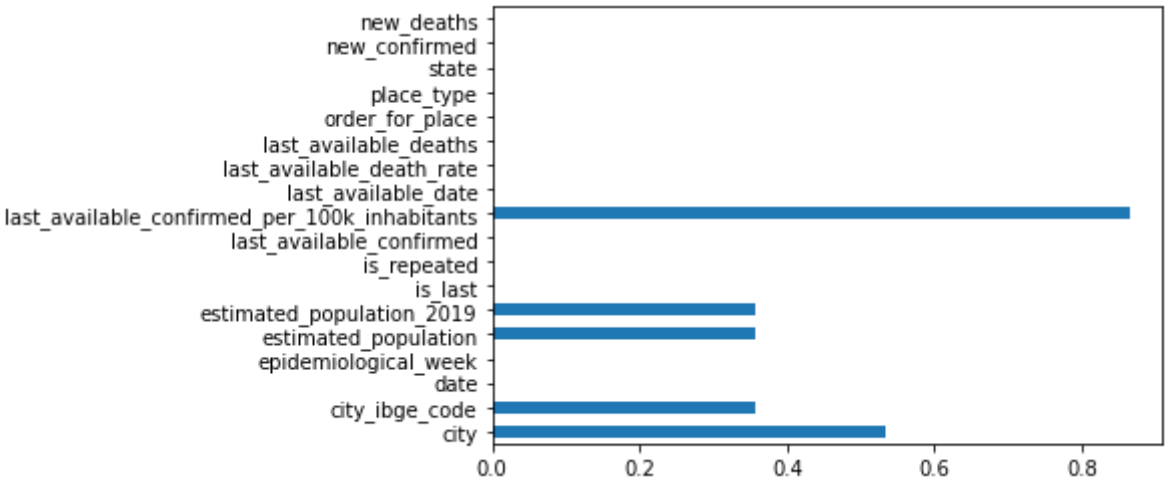


Figura 9: Quantidade de dados nulos - Base Brasil.IO.

Fonte: Autores

Ao fazer análise de outlier com o boxplot da figura 10, notamos que alguns códigos IBGE estavam com apenas dois dígitos e identificamos que essas linhas representam os dados totais do Estado por dia.

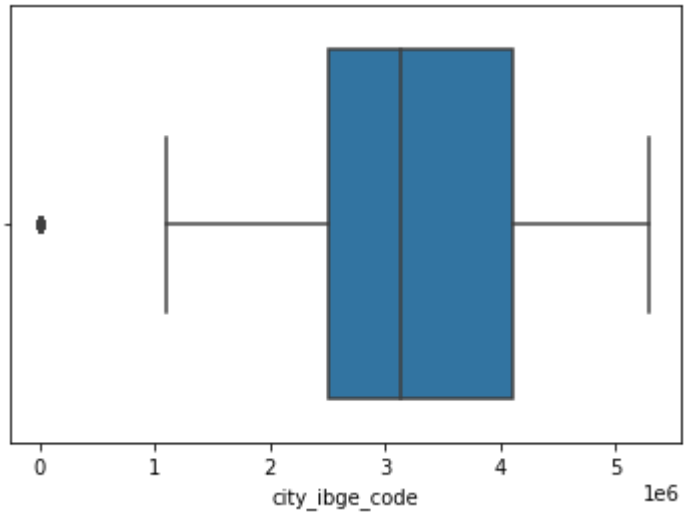


Figura 10: Boxplot do atributo city_ibge_code - Base Brasil.IO.

Fonte: Autores

Após analisar a distribuição de novos casos ao longo das semanas epidemiológicas, como mostra a figura 11, é possível notar a presença de valores negativos que, após uma análise, foi concluído que se tratam de casos remanejados entre Municípios. A Figura também chama atenção para alguns outliers como o ponto com mais de 100 mil casos que representa o

total de casos do dia 18/09/2021 no Rio de Janeiro. O outro ponto com mais de 60 mil casos representa o total de casos do dia 23/07/2021 no Rio Grande do Sul. É importante ressaltar que cada ponto representa um município brasileiro.

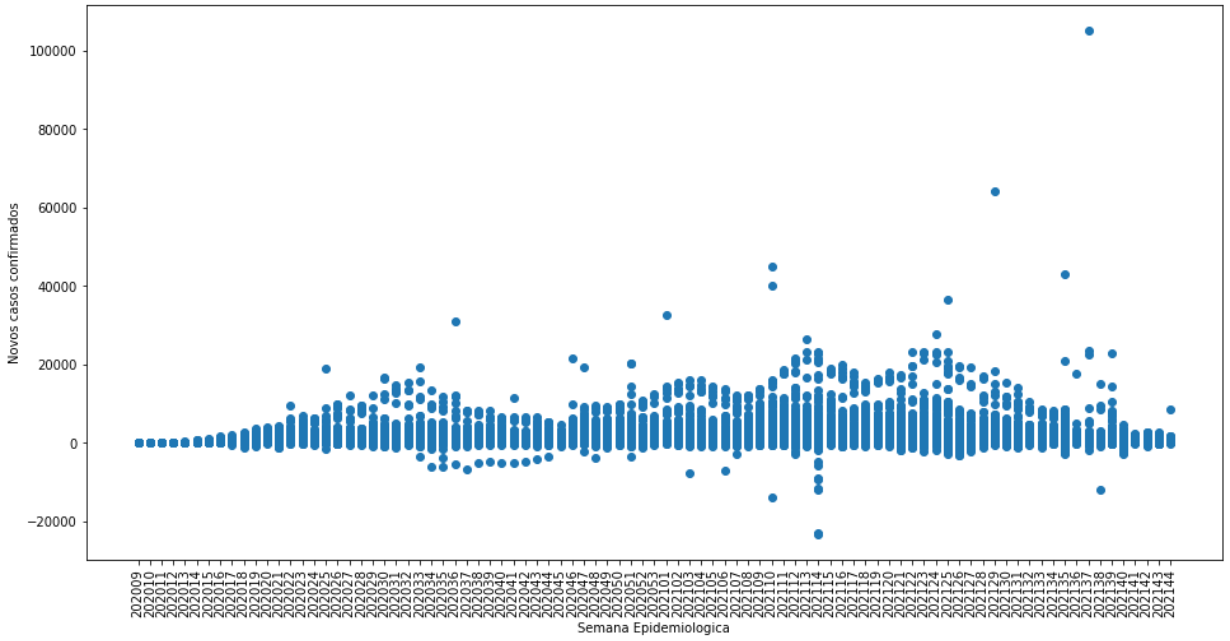


Figura 11: Distribuição de novos casos ao longo das semanas epidemiológicas.

Fonte: Autores

- IBGE

A tabela de dados sociodemográficos do IBGE possui dados como os códigos IBGE de 6 e 7 dígitos, nome do Município, Unidade da Federação, colunas com cada região do Brasil (Norte, Nordeste, Sul, Sudeste e Centro Oeste), população até 2020, densidade demográfica, estimativa até 2020 do percentual da população com 60 anos ou mais, estimativa até 2020 do percentual da população do sexo masculino, percentual da população em área urbana até 2010, percentual da população de cor/raça branca até 2010, PIB e PIB per capita de 2018, Índice de Gini da renda domiciliar per capita dos Municípios até 2010, quantidade de postos de trabalho de profissionais médicos, no SUS e em hospitais privados em 2019 de acordo com o Cadastro Nacional de Estabelecimentos de Saúde (CNES), quantidade de leitos hospitalares, exceto os leitos psiquiátricos, no SUS e em hospitais privados (dezembro de 2019) de acordo com o Cadastro Nacional de Estabelecimentos de Saúde (CNES) e percentual da população com pelo menos nível superior de graduação concluído (2010).

As únicas colunas com dados nulos são a de leitos (cerca de 35%), postos de trabalho (cerca de 4%), população com 60 anos ou mais (0,36%), percentual de população branca (0,14%), percentual da população em área urbana (0,14%) e Gini (0,09%). Analisando a

variável de população, foi feita uma visualização gráfica para identificar os 10 Estados brasileiros com a maior quantidade populacional. É possível notar na Figura 12 que São Paulo lidera em quantidade de população, seguido de Minas Gerais e Rio de Janeiro.

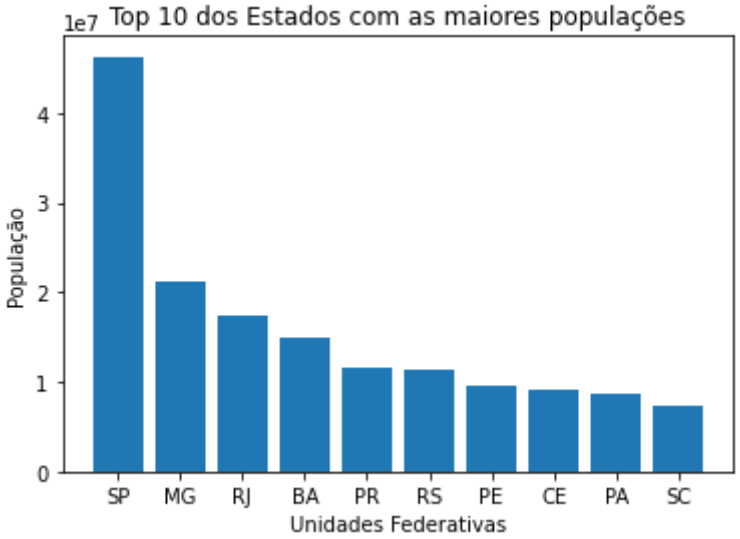


Figura 12: População por Estado.

Fonte: Autores

### 3.3.2 Pré-Processamento

Para que o modelo possa responder aos dados imputados de forma eficiente, realizamos algumas etapas de pré-processamento como alteração da granularidade através de agrupamentos, *feature selection, join* a partir das colunas, *feature engineering* e remoção de dados faltantes ou insuficientes. Tal etapa foi essencial para o treinamento do modelo escolhido, porém sua realização não foi sequencial. Por vezes encontramos a necessidade de retomar a manutenção dos dados a partir dos erros provenientes da execução do modelo. Entende-se que tal ação é natural do pipeline de Ciência de Dados.

Como já mencionado, o grande volume dos dados do SIPNI exige grande capacidade computacional para serem processados, além disso, esses dados são disponibilizados a nível de indivíduo e não é nosso objetivo realizar uma análise neste nível de detalhe, mas sim a partir dos Municípios. Com isso, utilizamos o SQL (Anexo III) para criar novas tabelas com dados agrupados dos Estados selecionados (RJ, SP, SC, MS, PA, AP e RR), considerando apenas algumas colunas para realizar o agrupamento: *estabelecimento_municipio_codigo, estabelecimento_municipio_nome, estabelecimento_uf, paciente_enumsexobiologico, paciente_racacor_valor, vacina_categoria_nome, vacina_grupoatendimento_nome,*

*vacina_descricao_dose*, e *vacina_nome*. Além disso, agrupamos a coluna de *vacina_dataaplicacao* pela semana epidemiológica disponibilizada no site do SINAN[38] (Sistema de Informação de Agravos de Notificação).

A partir disso, foi realizada uma etapa de feature engineering[39], que consiste na adição de novas colunas com dados provenientes da aplicação de alguma manipulação numérica ou categórica nos dados dos atributos existentes. Isso adiciona ao dataset features que podem descrever mais adequadamente características do problema de pesquisa investigado, por consequência se adequando melhor ao modelo e apresentando melhores resultados. Aplicamos essa engenharia de características ainda no PostgreSQL, criando um campo calculado que trazia a quantidade de vacinados de acordo com o agrupamento explicado no parágrafo anterior. Após levar os dados para o Python foi gerada a coluna de quantidade acumulada de vacinados através da biblioteca Pandas, acumulando apenas pelo Município e pela Semana Epidemiológica. Foi criada ainda uma coluna de cobertura vacinal utilizando a coluna com quantidade acumulada de pessoas vacinadas dividida pela população estimada de cada município, ou seja, tal coluna deve representar o percentual da população que foi vacinada naquela cidade.

Realizar uma junção (join) numa base de dados significa integrar tabelas que possuem informações relacionadas entre si, a partir de colunas que possuem dados em comum. Após a análise exploratória na base intermediária citada no parágrafo anterior, entendemos que para o projeto era necessário juntar informações das demais bases à base resultante do tratamento SIPNI, realizando a junção para o mesmo código IBGE do município e na mesma semana epidemiológica. As colunas adicionadas foram *new_confirmed* (novos casos confirmados de COVID-19) e *new_deaths* (novos óbitos por COVID-19), da base caso_full e as colunas PIB per capita, Gini, População e Código IBGE de 7 dígitos, da base Dados_Sociodemográficos. Essa ação foi reproduzida para cada uma das Unidades Federativas selecionadas, o que resultou em 7 conjuntos de dados.

O resultado deste processamento foi uma tabela com 11 colunas, sendo elas: *estabelecimento_municipio_codigo, estabelecimento_municipio_nome, sem_epidem_ano, qt_vacinados, qt_vacinados_acum, PIB_P_CAP, GINI, POP, city_ibge_code, cobertura_vacinal, new_confirmed* e *new_deaths*.

---

[38]http://portalsinan.saude.gov.br/calendario-epidemiologico-2020/43-institucional/171-calendario-epidemiologico-2021

[39] https://www.kaggle.com/ryanholbrook/what-is-feature-engineering

### 3.3.3 Modelagem

Para realizar a análise do impacto da vacinação contra o COVID-19 na mortalidade em decorrência de COVID-19 a nível municipal, precisamos entender como os demais indicadores da doença são afetados quando comparados aos indicadores da campanha de vacinação. Definida a questão de pesquisa, ("Qual o impacto da vacinação contra COVID-19 na mortalidade por COVID-19 nos municípios do Brasil?"), entendemos que os indicadores apropriados para realizar esta análise devem evidenciar de forma relevante essas informações a nível municipal.

A partir daí, os indicadores selecionados para a construção do modelo foram a cobertura vacinal (em termos percentuais) e o número de óbitos nos Municípios no decorrer das semanas epidemiológicas. Tal decisão foi tomada segundo os seguintes critérios:

- Utilizar a quantidade acumulada de pessoas vacinadas pode enviesar o modelo, uma vez que valores absolutos podem resultar em diferentes comportamentos devido a fatores como o tamanho da população, limitando comparações entre os Estados ou entre os Municípios. Dessa forma, a cobertura vacinal - quantidade acumulada de vacinas aplicadas pela população - acaba sendo uma medida mais justa e adaptável ao modelo, fazendo com que todos os municípios compartilhem da mesma escala.
- O pior cenário resultante da infecção pelo Coronavírus é a morte do indivíduo infectado. Neste sentido, entende-se que a maior contribuição da Campanha de Vacinação contra a doença seja na redução do número de óbitos.

O número de óbitos nos Municípios é definido como um dado de contagem. Dados de contagem são o registro de eventos que ocorrem com uma certa frequência de forma a apresentar um valor diferente a cada observação. Tadano et al, (2009) citam que dados de contagem são analisáveis a partir de Modelos Lineares Generalizados (MLGs), mais especificamente por Regressões de Poisson. Tomando essa informação como premissa, escolhemos a Regressão de Poisson como a técnica a ser aplicada, de forma a entender se há relação entre a cobertura vacinal e o número de óbitos ao longo das semanas epidemiológicas, uma vez que a Regressão de Poisson é utilizada para modelar dados de contagem, se adequando assim aos dados do presente trabalho.

Segundo Gujarati e Porter [2011], o modelo de Regressão de Poisson é explicado pela seguinte equação:

$$Y_i = E(Y_i) + u_i = \mu_i + u_i$$

onde $u_i$ são os resíduos, ou seja, a diferença entre os valores observados e estimados de $Y_i$, sendo que cada Y é distribuído independentemente como variável aleatória de Poisson, com média $\mu_i$ para cada indivíduo expressa como:

$$\mu_i = E(Y_i) = \beta_1 + \beta_2 * X_{2i} + \beta_3 * X_{3i} + \ldots + \beta_k * X_{ki}$$

onde cada X representa uma variável explicativa, que, no caso do presente trabalho, é somente a cobertura vacinal.

A técnica de regressão foi experimentada em diversos cenários, de forma a perceber em quantas semanas o impacto da vacinação no número de óbitos é maior. Escolhemos dessa forma o seguinte método: Estimar qual lag (defasagem) de semanas aplicado ao número de mortes em um município resulta em um modelo de melhor qualidade, ou seja, qual lag melhor adequa a evolução dos óbitos semanais à variável explicativa cobertura vacinal semanal, ao longo de todo o período de vacinação de cada município. A qualidade do modelo é medida a partir dos critérios de informação AICc e Pseudo R², sabendo que, para isso, o modelo tem que minimizar o AICc e maximizar o pseudo R².

Critérios de informação são métricas de avaliação de modelos baseados em Teoria da Informação. O AICc é uma adequação do critério de Akaike (1973) para seleção de modelos com pequenas amostras. Desenvolvido por Hurvich & Tsai (1989), levam em conta o AIC e um ajuste dos parâmetros:

$$AICc = AIC + \frac{2*(p+1)*(p+2)}{n-p-2} = -2\sum_{i=1}^{n} ln\, L(\hat{\mu}_i, y_i) + 2p + \frac{2*(p+1)*(p+2)}{n-p-2}$$

onde $p$ é o número de parâmetros e $n$ o número de amostras.

O Pseudo-R²[40] é um critério de seleção de MLGs a partir da variância da variável dependente. Recebe esse nome pois o R² verdadeiro, utilizado para modelos lineares, varia entre 0 e 1. De forma geral, ele diz o quanto um modelo foi capaz de prever a variabilidade das informações. Dessa forma, quanto maior for seu valor, maior é a previsibilidade das variâncias do modelo. Assim como o AIC, existem diversos métodos para se calcular o Pseudo-R². Este trabalho utiliza para dados discretos o Pseudo-R² de McFadden's:

$$Pseudo\, R^2 = 1 - \frac{ln\, L(Ms)}{ln\, L(Mt)}$$

[40] https://betaanalitica.com.br/existe-r%C2%B2-para-modelos-lineares-generalizados-glms/

onde $L(Ms)$ é a probabilidade do modelo selecionado e $L(Mt)$, é do modelo treinado ter apenas um intercepto.

Para a aplicação do modelo foi utilizado o pacote estatístico em Python statsmodel[41], que fornece funções para estimativa de modelos estatísticos. Utilizando a classe MLG do pacote, foi invocada a opção Poisson() para o atributo de famílias de regressões possíveis. Também foi utilizado o pacote Patsy[42], que é utilizado para descrever modelos estatísticos.

Para buscar o lag citado anteriormente, foi preciso realizar a construção de uma nova coluna chamada *mortes_lag*. Tal coluna representa o número de semanas decorridas desde que o município alcançou uma determinada cobertura vacinal até a semana em que os óbitos são contabilizados. O range do lag foi variado de 0 a 7 semanas, de forma a manter uma quantidade de amostras suficientes para treinar o modelo.

Tendo em vista que  não é o objetivo realizar previsões (e, portanto, não haveria necessidade de um conjunto específico para teste do modelo), não foi necessário dividir o dataset original em treinamento (para aprender o modelo de regressão) e teste (para avaliar o modelo aprendido). Portanto, a função de treinamento recebeu o conjunto de dados em sua totalidade.

O próximo passo é definir no Python uma variável que vai receber a expressão de regressão em notação Patsy. Tal expressão serve para configurar os vetores X e Y que são passados à função  para treinamento:

```
expr = """mortes_lag ~  cobertura_vacinal"""
y_train, X_train = dmatrices(expr, df_train, return_type='dataframe')
poisson_training_results = sm.GLM(y_train, X_train, family=sm.families.Poisson()).fit()
```

Figura 13: Expressão de Regressão em notação Patsy e função de treinamento do modelo.
Fonte: Autores

Inicialmente o treinamento do modelo apresentou alguns erros provenientes da dispersão dos dados referentes a  quantidade de mortes no decorrer das semanas. Para tratar desses casos, ao rodar modelo, identificamos quais cidades apresentavam tais características e as removemos da lista de cidades apresentadas para o treinamento. Dentre as 1200 cidades analisadas, 22 foram retiradas.

[41] https://www.statsmodels.org/stable/index.html
[42] https://patsy.readthedocs.io/en/latest/overview.html

Em seguida, realizamos o fit do modelo como mostrado na figura 12 (ou seja, aplicamos a técnica de regressão e avaliamos seu resultado), para cada município e cada lag n.

Os modelos estatísticos do pacote statsmodel possuem atributos que são gerados automaticamente após o treinamento do modelo. Dessa forma, foi possível gerar um dataframe contendo o código do município, o coeficiente de regressão e os critérios de avaliação do modelo, dentre eles o AICc. Os dados citados podem ser visualizados na tabela 3:

| lag | cod_ibge_mun | coef_cob_vac | aic | aicc | bic | pseudo_r | pvalue |
|---|---|---|---|---|---|---|---|
| 1.0 | 3557303.0 | -0.014586 | 79.157225 | 79.520862 | -94.950080 | 0.164758 | 0.023521 |
| 2.0 | 3557303.0 | -0.017081 | 76.649368 | 77.024368 | -92.384007 | 0.193235 | 0.016343 |
| 3.0 | 3557303.0 | -0.018584 | 74.648537 | 75.035633 | -87.901889 | 0.196785 | 0.017080 |
| 4.0 | 3557303.0 | -0.020768 | 72.375584 | 72.775584 | -83.723040 | 0.206992 | 0.016271 |
| 5.0 | 3557303.0 | -0.023577 | 69.967137 | 70.380930 | -79.711829 | 0.220883 | 0.014760 |
| 6.0 | 3557303.0 | -0.027552 | 67.219889 | 67.648461 | -76.072629 | 0.243582 | 0.012108 |
| 7.0 | 3557303.0 | -0.032410 | 64.624322 | 65.068767 | -73.755321 | 0.278510 | 0.008419 |

Tabela 3: Estatísticas geradas para os municípios de SP

Fonte: Autores

O próximo passo foi plotar histogramas para verificar a frequência dos valores de AICc e dos Pseudo $R^2$, conforme descrito na Seção 3.3.4.

### 3.3.4 Interpretação dos resultados

Os histogramas mencionados na etapa anterior podem ser visualizados nas figuras 14 e 15. A partir deles é possível concluir que o lag que melhor ajusta o modelo é o igual a 7. Ou seja, é possível afirmar a partir disso que o impacto da vacinação nos óbitos dos municípios analisados pode ser percebido com uma defasagem de 7 semanas.

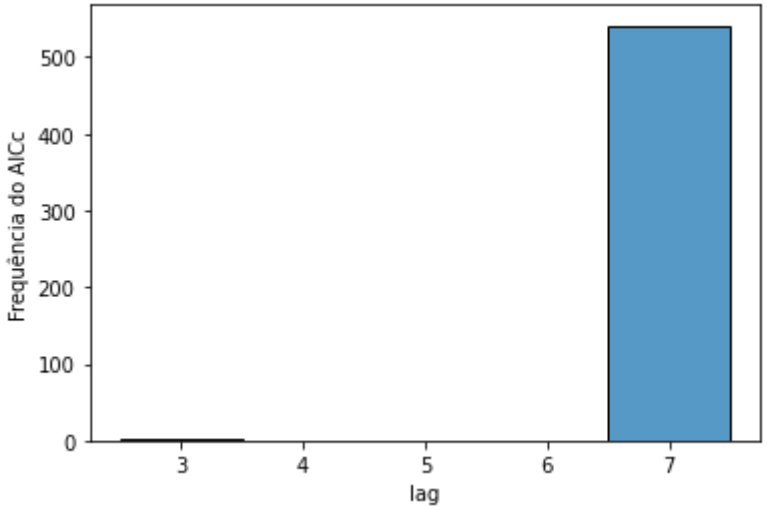


Figura 14: Lag que minimiza o AICc (exemplo dos dados de todos os municípios de SP)

Fonte: Autores

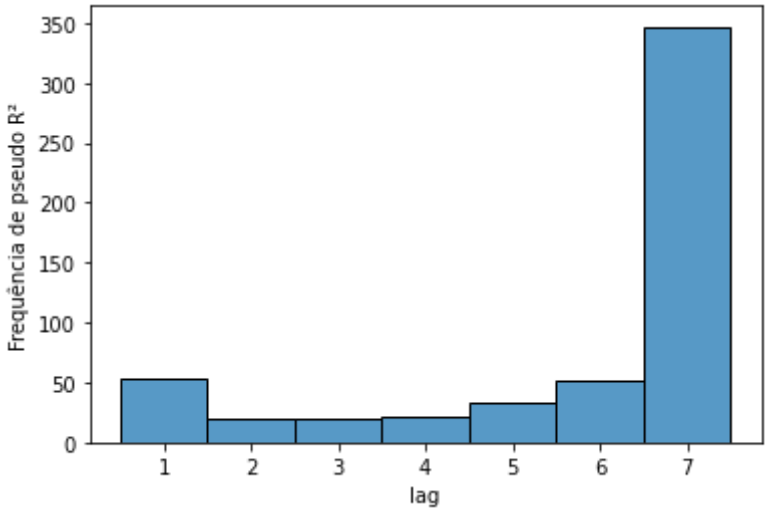


Figura 15: Lag que maximiza o Pseudo R² (exemplo dos dados de SP)

Fonte: Autores

Após a escolha do lag 7 para todos os municípios seguindo os critérios de minimizar o AICc e maximizar o Pseudo R², foi criada a coluna mortes_lag com os dados de novos óbitos defasados em 7 semanas para todos os municípios.

O coeficiente determina se uma alteração em uma variável preditora/explicativa (cobertura vacinal no nosso caso) torna a variável resposta (número de óbitos no nosso caso) mais ou menos provável. Coeficientes positivos tornam a variável resposta mais provável, já os coeficientes negativos a tornam menos provável [Selau et al. 2009]. Por exemplo, se o

coeficiente de cobertura vacinal for negativo, significa que uma alta cobertura vacinal está associada a valores menores na variável de resposta.

Já o p-value, determina se essa associação citada no parágrafo acima entre a variável resposta e a preditora é estatisticamente significativa. Para analisar o p-value, é preciso compará-lo com o nível de significância adotado. Um nível de significância de 0,05 indica uma probabilidade de 5% de rejeitar a hipótese H0: existe uma associação, quando ela é nula [Reis, 2008]. Em outras palavras, a probabilidade de concluir que há relação quando não existe. Se o p-value for menor do que o nível de significância, conclui-se que há uma associação estatisticamente significativa entre a variável de resposta e a preditora, caso contrário, não é possível concluir essa associação. Neste trabalho escolhemos o valor de 0,05 para o nível de significância.

Por questões de limitação de tempo para conclusão do trabalho, foram analisados os modelos gerados para os 10 Municípios com menores coeficientes e os 10 Municípios com maiores coeficientes para cada Estado para validar se de fato o avanço da vacinação impactou na redução do número de óbitos, e como este impacto pode ter variado entre municípios em diferentes estágios de cobertura vacinal. De modo a não ficar poluído, as figuras a seguir ilustram um exemplo dentre os municípios com maiores e menores coeficientes, para cada Estado.

A Figura 16 representa o município Francisco Morato que, segundo o modelo, possui um dos menores coeficientes (-0,062, $p < 0,001$), indicando que este local foi um dos mais impactados na redução do número de óbitos pelo aumento da cobertura vacinal quando olhamos o Estado de São Paulo. Já a figura 17 representa o município Santo Anastácio, que possui um dos maiores coeficientes de São Paulo (0,023, $p < 0,001$), indicando que o modelo interpretou que o aumento da vacinação acarreta no aumento do número de mortes; porém, é possível notar um pico de casos por volta da semana 31 de 2021, o que pode representar notificações de óbitos que estavam represadas. Dos 639 municípios de São Paulo analisados, 562 (87,9%) possuem coeficientes negativos, indicando assim que, para a maioria das Cidades, um maior número de vacinados leva a um menor número de óbitos.

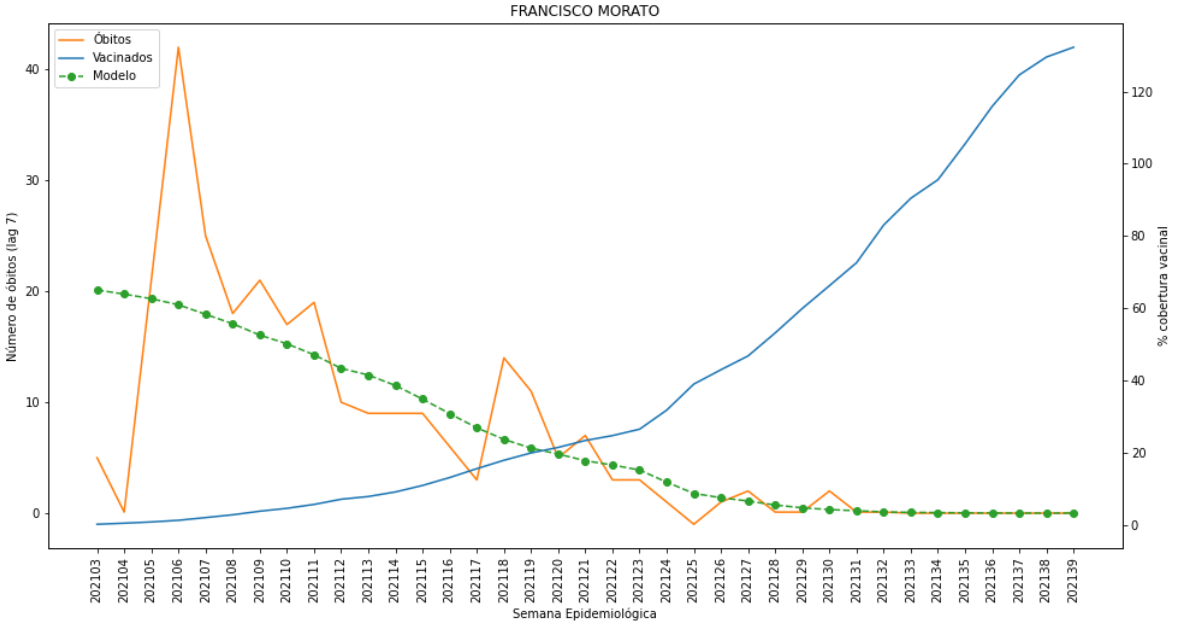


Figura 16: Município com um dos **menores** coeficientes de SP

Fonte: Autores

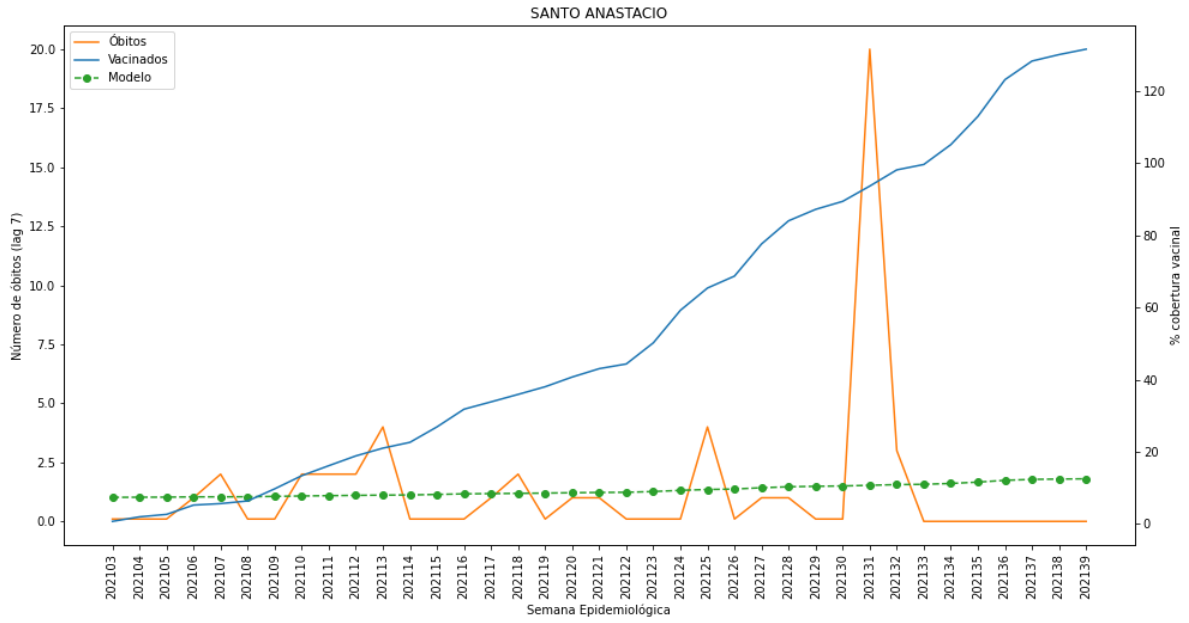


Figura 17: Município com um dos **maiores** coeficientes de SP

Fonte: Autores

Replicada a mesma análise para os demais Estados, como descrito a seguir.

Os municípios escolhidos do Rio de Janeiro foram Macaé (com coeficiente -0,045 e $p<0,001$ e Japeri (com coeficiente 0,038 e $p < 0,001$), que também indica pela figura um

possível dado represado na semana 32. Dos 92 municípios do Rio de Janeiro, 79 (85,9%) possuem coeficientes negativos.

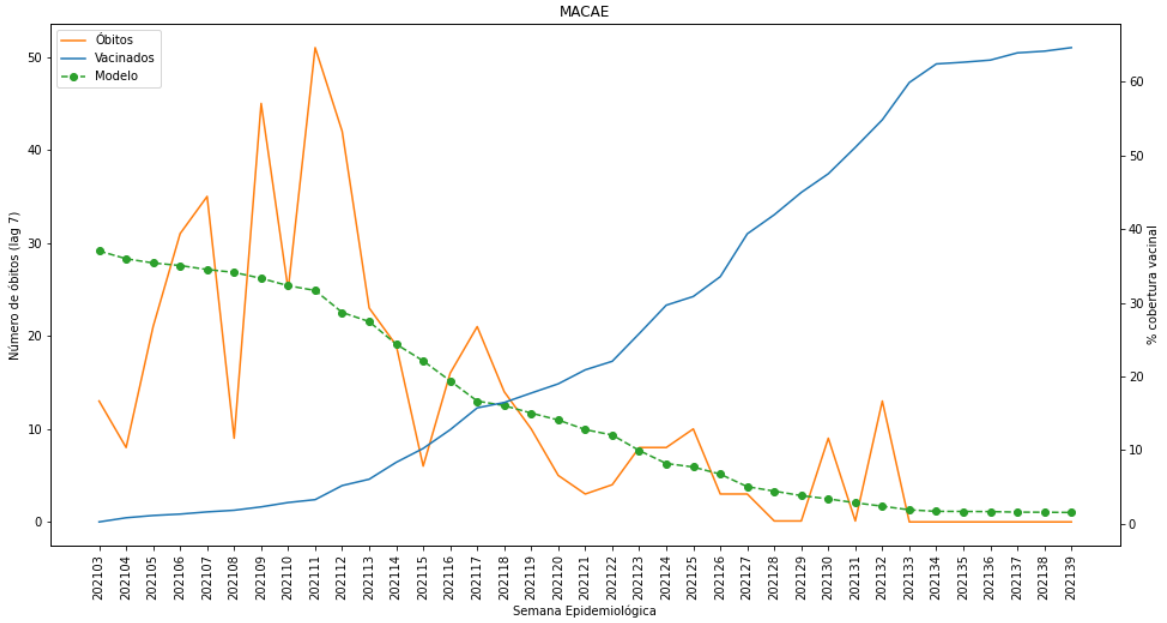


Figura 18: Município com um dos **menores** coeficientes do RJ

Fonte: Autores

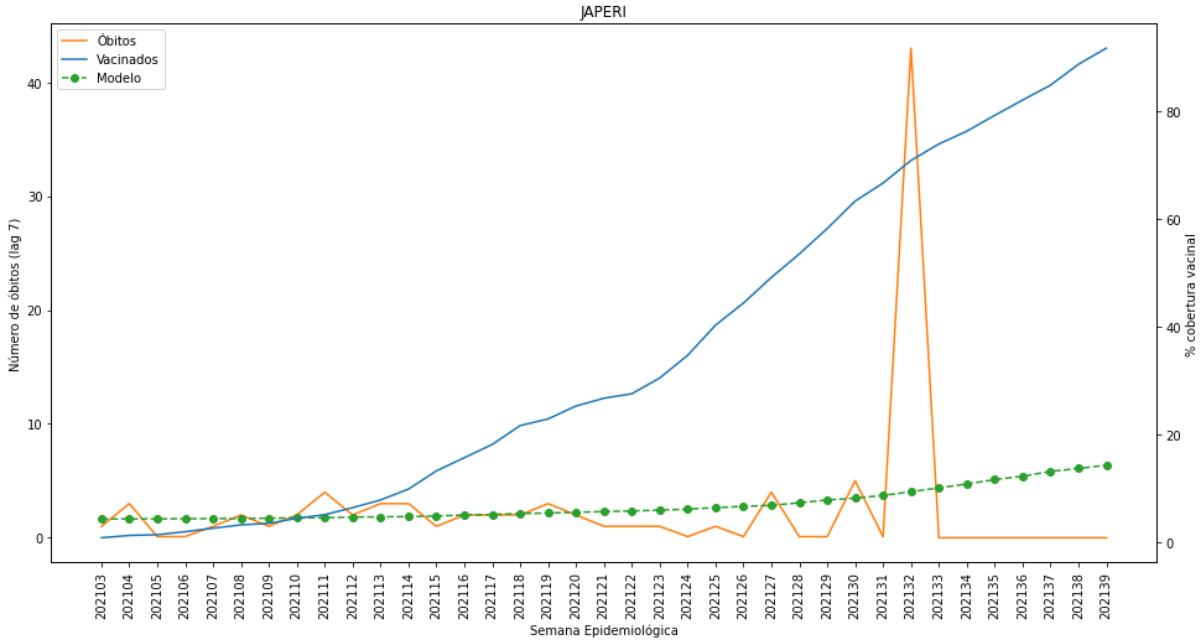


Figura 19: Município com um dos **maiores** coeficientes do RJ

Fonte: Autores

Para o Estado de Santa Catarina, ilustrado na Figura 20, o município de Florianópolis (que teve um coeficiente de -0,047 e $p < 0,001$) e na Figura 21 o município de Vitor Meireles

(com coeficiente 0,018 e p < 0,13). Esses resultados mostram que para o município de Vitor Meireles o modelo indica que a associação entre o número de óbitos e a cobertura vacinal não é estatisticamente significativa, uma vez que o p-value apresentou um valor superior ao nível de significância adotado (0,05). Dos 285 municípios de Santa Catarina, 263 (92,2%) apresentaram coeficiente negativo.

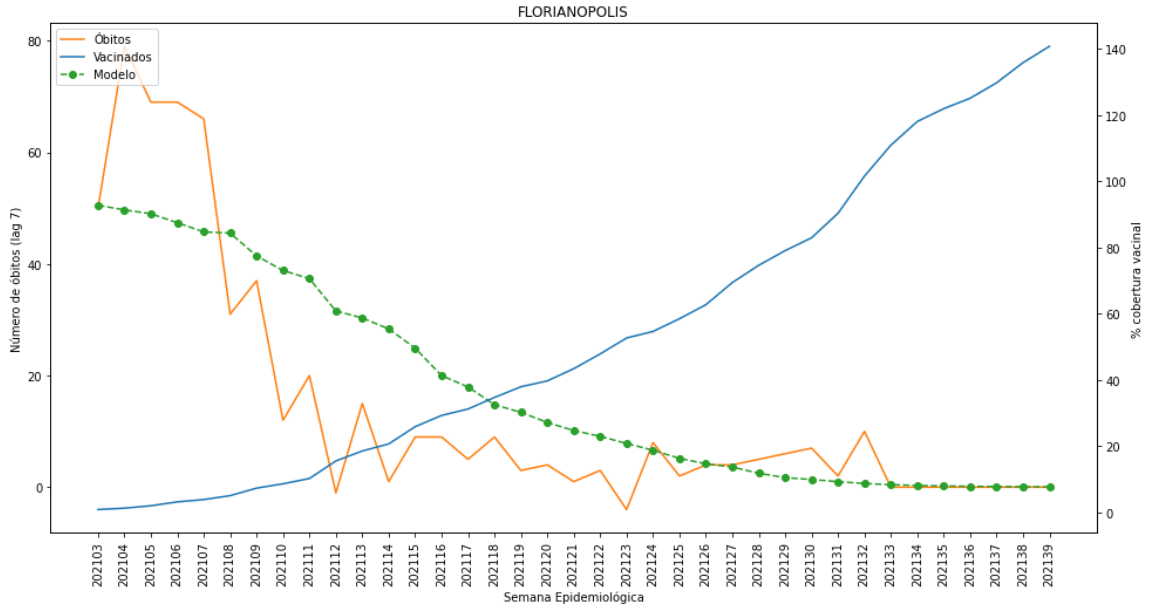


Figura 20: Município com um dos **menores** coeficientes de SC

Fonte: Autores

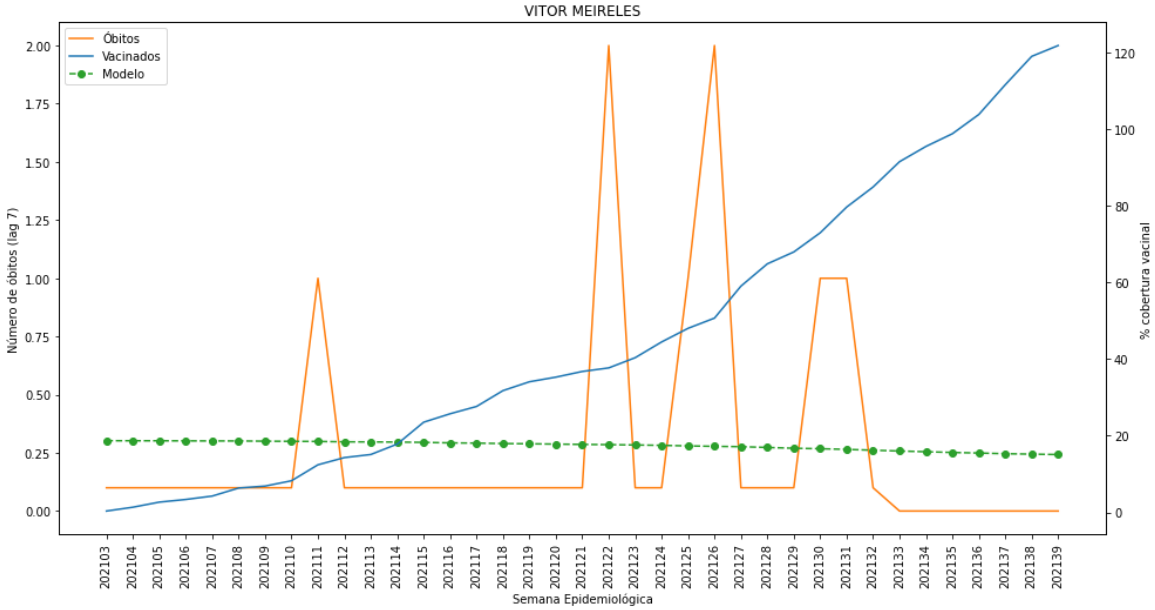


Figura 21: Município com um dos **maiores** coeficientes de SC

Fonte: Autores

Para o Estado do Mato Grosso do Sul, ilustrado na Figura 22, o município de Três Lagoas que teve um coeficiente de -0,047 e $p < 0,001$, e na Figura 23 o município de Paraíso das Águas com coeficiente de -0,003 e $p = 0,78$. De acordo com o resultado, o modelo referente ao município Paraíso das Águas não indica uma associação estatisticamente significativa entre as variáveis analisadas. Dos 79 municípios analisados referentes ao Mato Grosso do Sul, 78 apresentaram coeficiente negativo (98,7%).

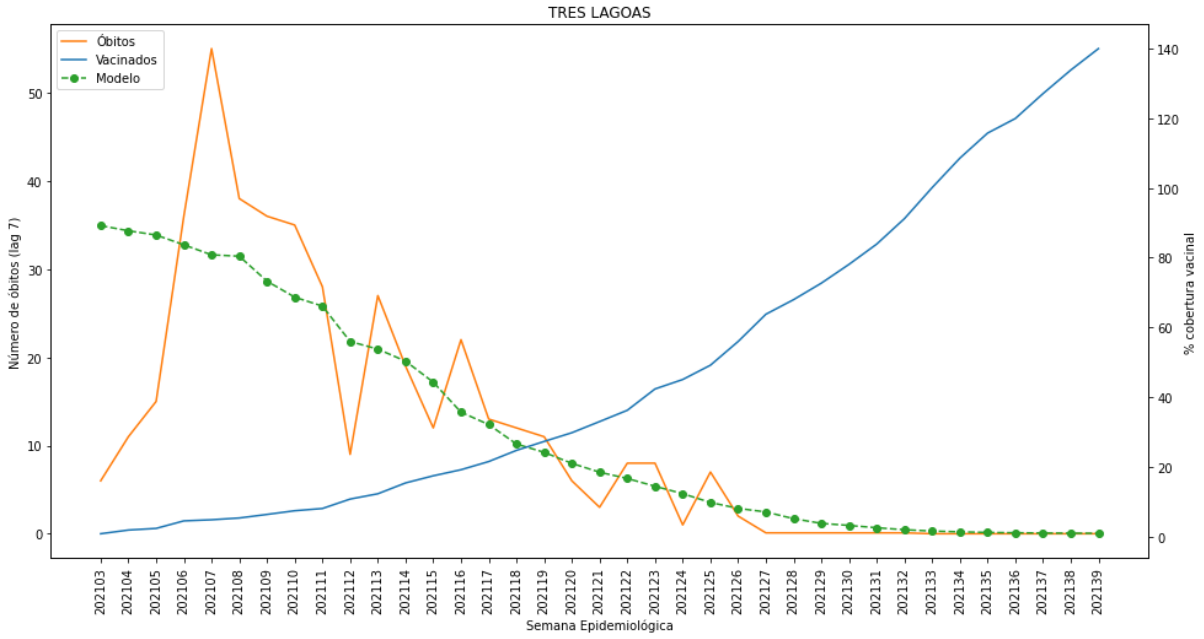


Figura 22: Município com um dos **menores** coeficientes do MS

Fonte: Autores

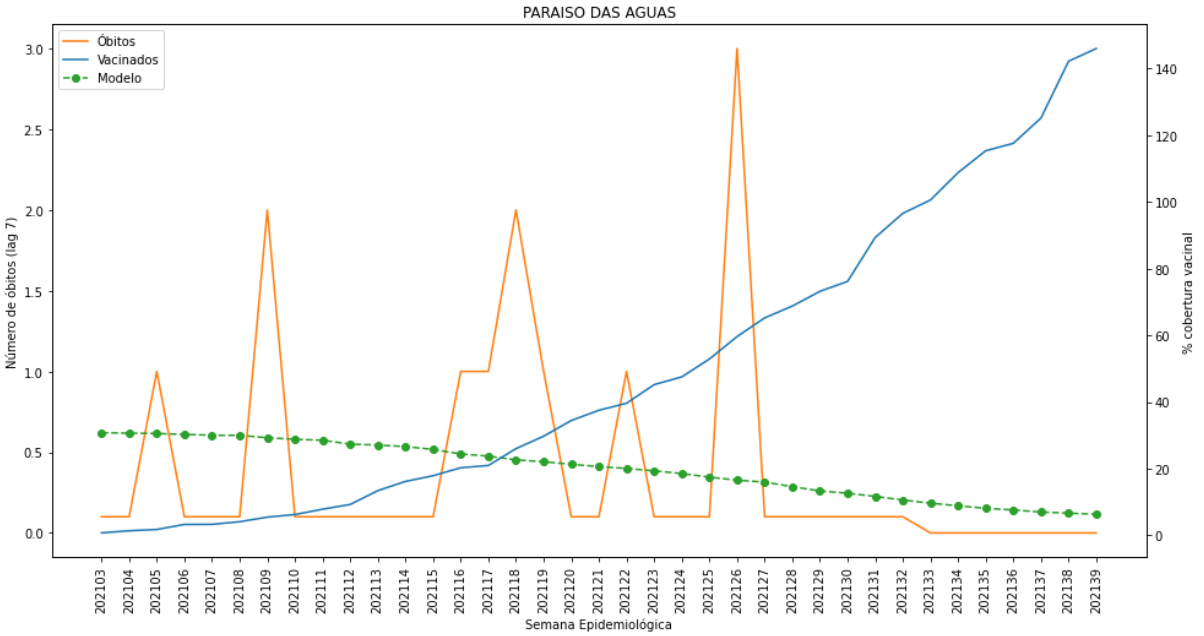


Figura 23: Município com um dos **maiores** coeficientes do MS

Fonte: Autores

Para o Estado do Pará, ilustrado na Figura 24, o município de Vigia que teve um coeficiente de -0,17 e $p < 0,001$, e na Figura 25 o município de São Geraldo do Araguaia com coeficiente 0,012 e $p = 0,3$. De acordo com o p-value, não há associação estatisticamente significativa entre o número de óbitos e a cobertura vacinal para o município de São Geraldo do Araguaia. Dos 143 municípios analisados pertencentes ao Pará, 131 (91,6%) apresentaram coeficiente negativo.

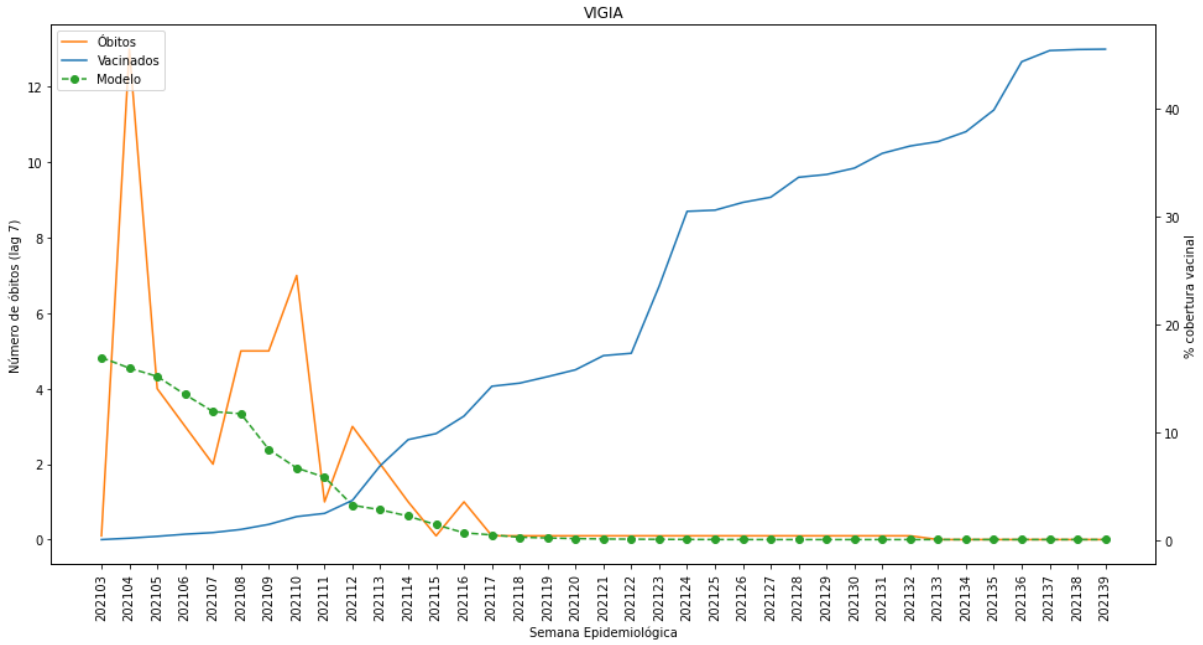


Figura 24: Município com um dos **menores** coeficientes do PA

Fonte: Autores

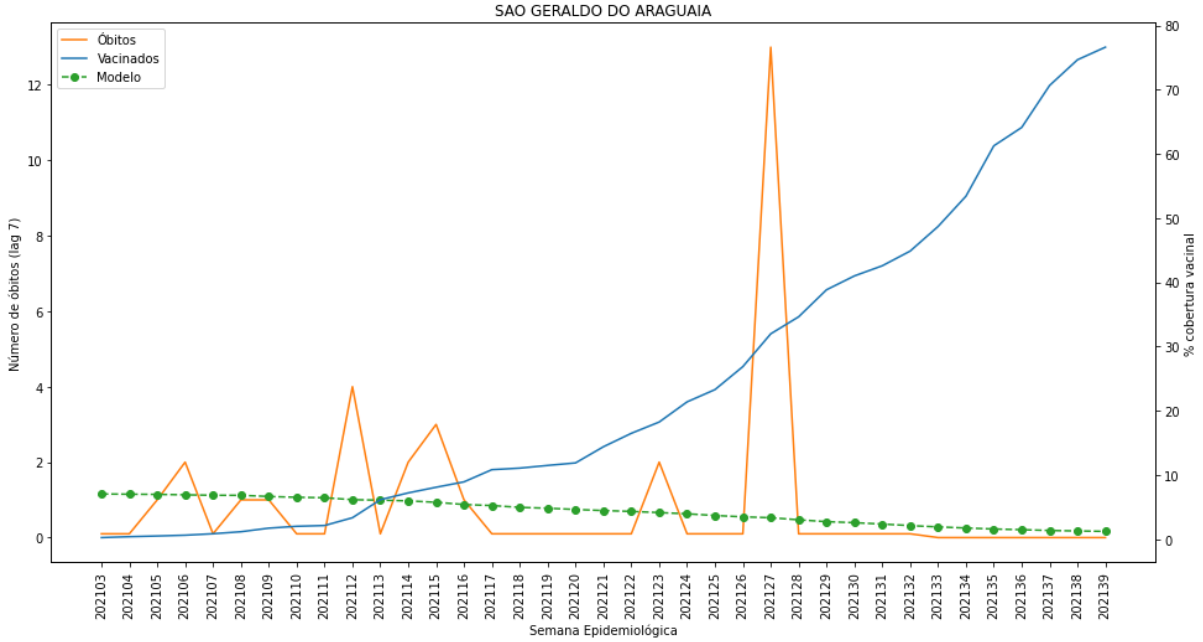


Figura 25: Município com um dos **maiores** coeficientes do PA

Fonte: Autores

Para o Estado do Amapá, ilustrado na Figura 26, o município de Macapá que teve um coeficiente de -0,05 e $p < 0,001$, e na Figura 27 o município do Amapá com coeficiente 0,022

e p = 0,3, indicando que este município não apresenta associação estatisticamente significativa entre o número de óbitos e a cobertura vacinal. Dos 11 municípios analisados pertencentes ao Amapá, 8 apresentaram coeficiente negativo (72,7%).

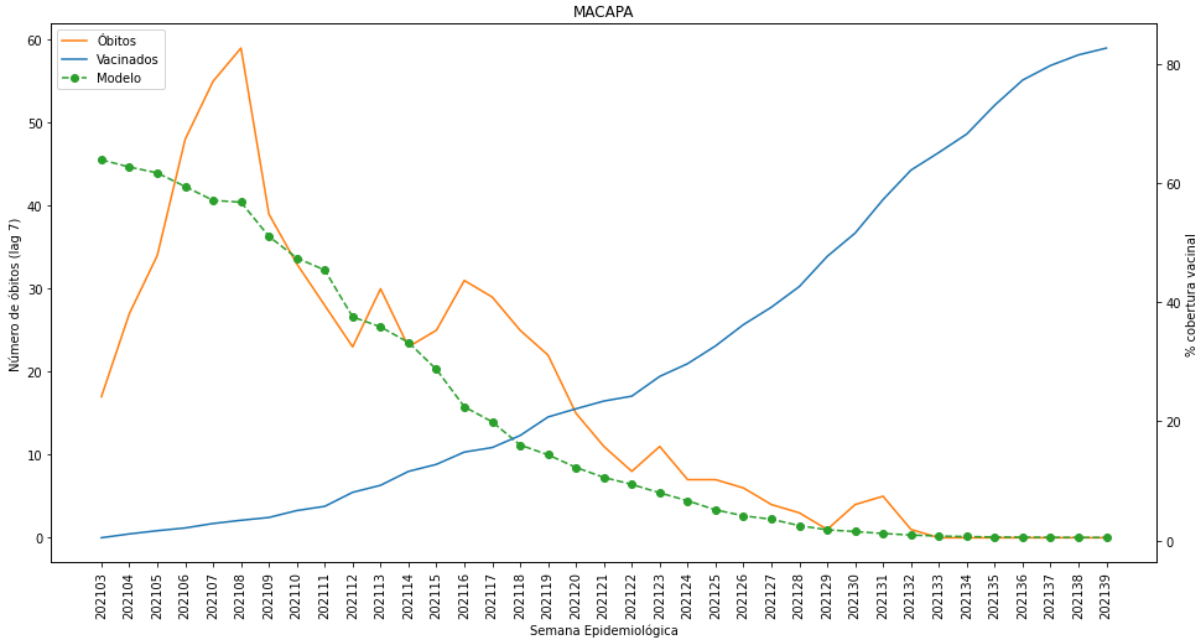


Figura 26: Município com um dos **menores** coeficientes do AP

Fonte: Autores

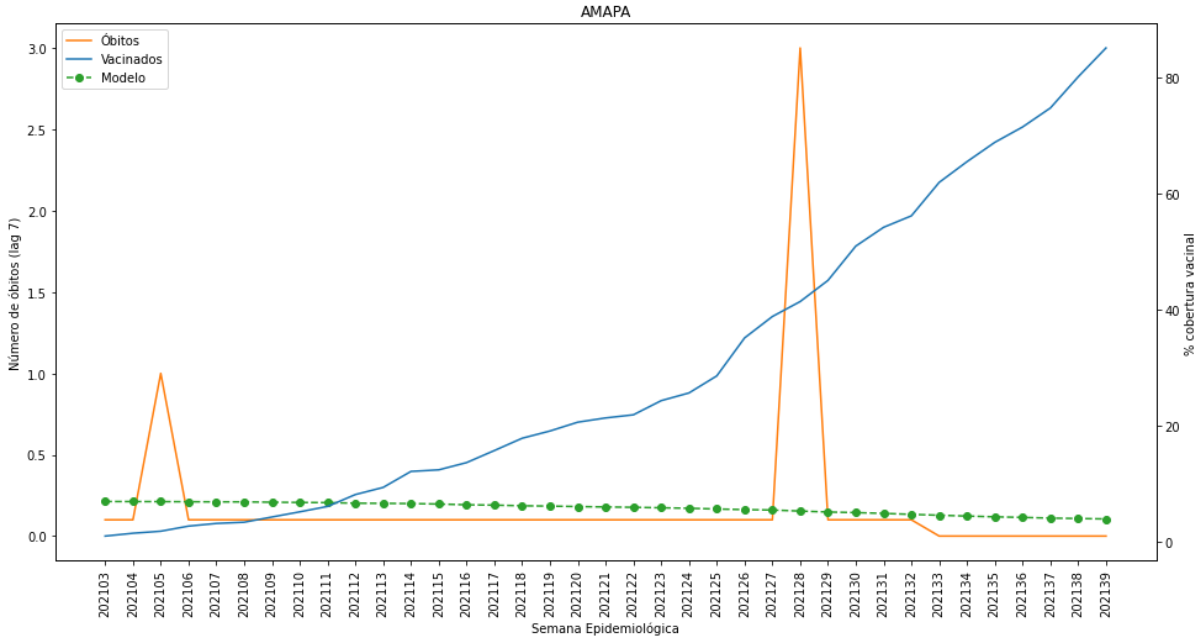


Figura 27: Município com um dos **maiores** coeficientes do AP

Fonte: Autores

Para o Estado de Roraima, ilustrado na Figura 28, o município de Boa Vista que teve um coeficiente de -0,033 e p < 0,001, e na Figura 29 o município de Pacaraima com coeficiente 0,022 e p = 0,15. É possível concluir que, de acordo com o p-value superior ao nível de significância para Pacaraima, não há associação estatisticamente significativa entre as variáveis analisadas desse município. Dos 15 municípios analisados pertencentes a Roraima, 10 apresentaram coeficiente negativo (66,7%).

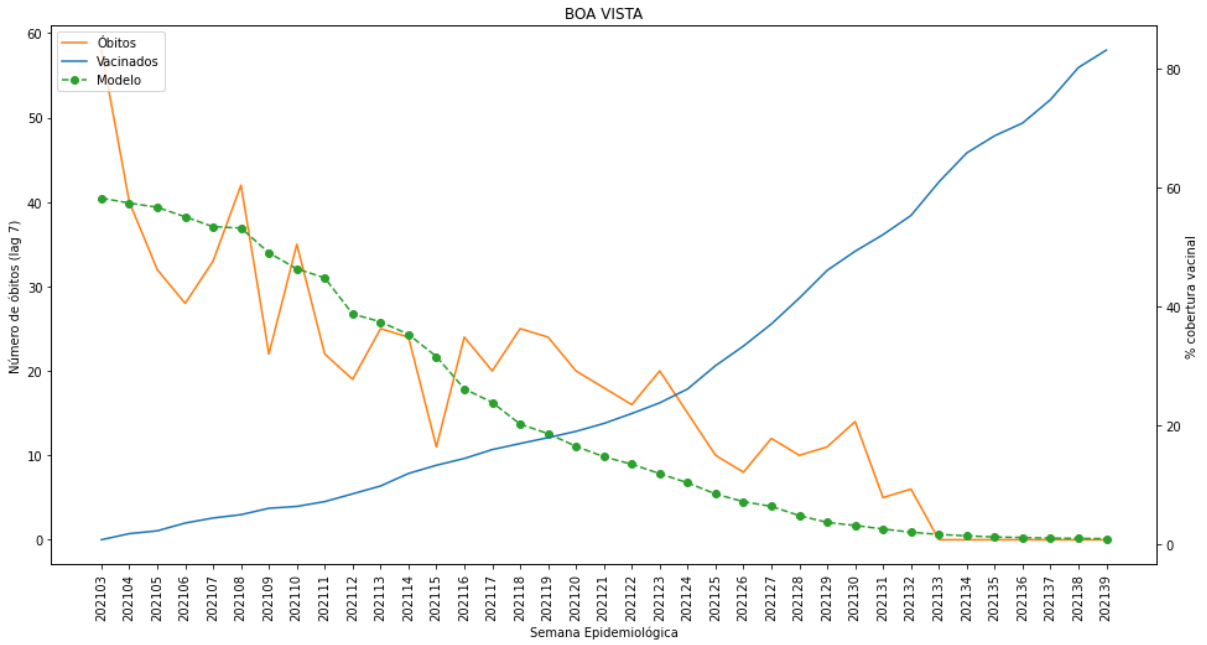


Figura 28: Município com um dos **menores** coeficientes de RR

Fonte: Autores

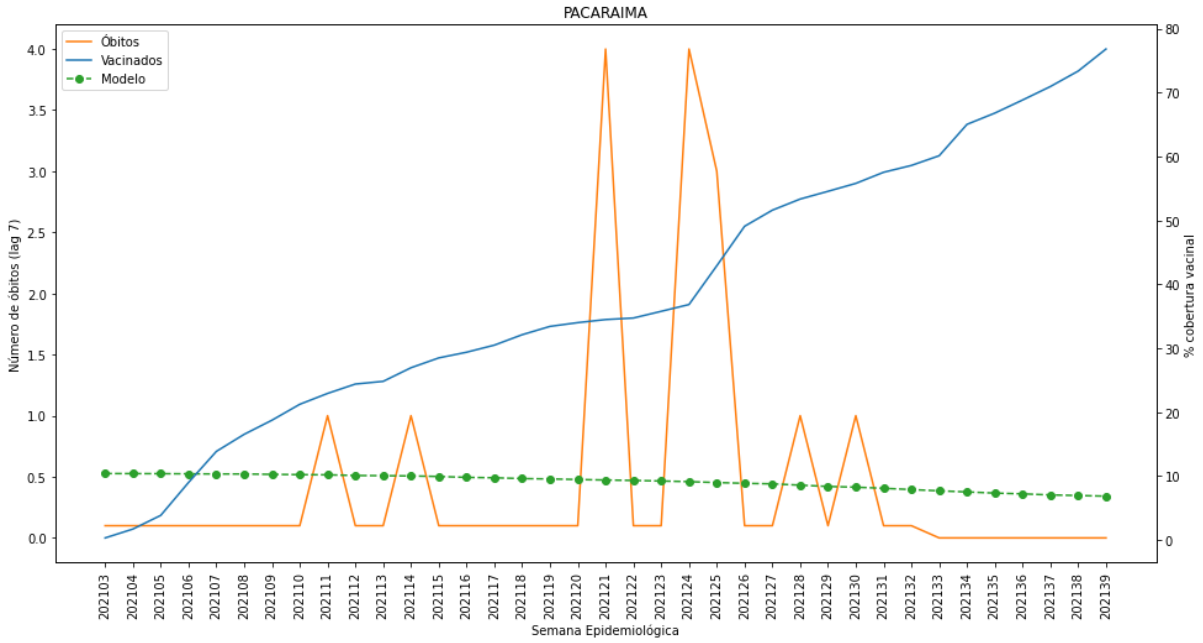


Figura 29: Município com um dos **maiores** coeficientes de RR

Fonte: Autores

É possível notar nas figuras que, para alguns municípios, a cobertura vacinal ultrapassa 100%, isso pode ser explicado por dois motivos: primeiro, os dados de população não são precisos, já que foi utilizada uma estimativa de 2020 e o último censo IBGE ocorreu em 2010; além disso, foi considerada a informação de municípios referentes ao estabelecimento de vacinação, ou seja, o paciente não necessariamente irá tomar a vacina no município de residência. Essa escolha, do município do estabelecimento, se deu pois a coluna de Município de residência continha cidades de diversos Estados diferentes que fugia do escopo dos 7 Estados selecionados.

# 4 CONCLUSÃO

O volume de dados gerados a partir da pandemia de COVID-19 foi suficientemente grande e consequentemente acarretou numa gama de opções para estudo e análise destas informações. Além disso, é notoriamente crescente a profusão de ambientes de apoio computacional para iniciativas de Ciência de Dados, incluindo plataformas independentes de domínio especificamente voltadas para Ciência de Dados, Sistemas de Gestão de Workflow Científicos, e ambientes genéricos de programação com um número crescente de bibliotecas para apoio às atividades de Ciência de Dados.

Neste contexto, aplicar metodologias e técnicas de Ciência de Dados mostrou-se como uma boa opção para entender tais dados, obter insights e extrair conhecimento útil que agregue valor à tomada de decisões de gestores públicos com relação à contenção da evolução da pandemia de COVID-19 no mundo todo, em especial no Brasil, um país de dimensão continental e com desigualdades profundas cuja compreensão da influência na gestão da evolução da pandemia ainda é um desafio. Em particular, o presente trabalho aplicou o ciclo de vida de um projeto de Ciência de Dados no domínio da vacinação da COVID-19 nos municípios brasileiros, utilizando as ferramentas mais utilizadas no mercado e no meio acadêmico, de forma a reproduzir ao máximo a qualidade que esta metodologia propõe, com o propósito de analisar o impacto da vacinação nos óbitos por COVID-19, em diversos municípios brasileiros.

Ao longo da realização do trabalho, diversas plataformas de apoio computacional à Ciência de Dados foram estudadas, e algumas foram avaliadas quanto ao uso e suporte adequado ao grande volume de dados manipulado. Foi possível perceber que, até mesmo para um grande volume de dados, existem alternativas eficientes de manipulação de dados, que podem trazer resultados satisfatórios dependendo do grau de exigência dado ao propósito do projeto. Da mesma forma, a aplicação de técnicas de modelagem de dados (em especial regressão) gerou insights a partir dos resultados, de forma a levantar algumas hipóteses que endereçam a questão de pesquisa investigada. Por outro lado, alguns ambientes computacionais - que a princípio eram muito promissores - impuseram limitações que impediram seu uso para o propósito deste projeto.

Após a etapa de modelagem dos dados e extração de conhecimento (Mineração dos Dados), na qual técnicas de regressão foram avaliadas experimentalmente em cenários que variaram a defasagem entre a semana correspondente à cobertura vacinal e a semana que maximizou a percepção do impacto na quantidade de óbitos no mesmo município, foi possível

concluir que, na maior parte dos municípios considerados e considerando as premissas estabelecidas pela etapa de pré-processamento dos dados, a quantidade de óbitos por COVID-19 pode ser explicada em função da evolução da cobertura vacinal diária no mesmo município, através de técnicas de Regressão de Poisson, quando utilizada uma defasagem de 7 semanas e que, na maioria dos municípios, essa quantidade de óbitos segue tendência de queda com o aumento da cobertura vacinal.

O presente estudo foi realizado no contexto do NOIS (Núcleo de Operações e Inteligência em Saúde)[43], um grupo formado por alunos, professores e pesquisadores do Departamento de Engenharia Industrial e do Instituto Tecgraf da PUC-Rio, Fiocruz, USP e IDOR, que vêm pesquisando e desenvolvendo soluções inovadoras para monitoramento da COVID-19 e outros temas relacionados a Saúde, provendo evidências para decisões baseadas em dados. Os artefatos (código-fonte, datasets e metadados, visualizações geradas e o presente documento) gerados ao longo deste trabalho estão disponíveis publicamente em um repositório do Github, acessível neste link: https://github.com/yasmin-amaro/TCC-Yasmin-e-Sergio.

Trabalhos futuros podem expandir a análise realizada para outros municípios, e avaliar a influência de características sócio-demográficas dos municípios no impacto da cobertura vacinal em termos de óbitos, internações ou casos de COVID-19, além disso, utilizar outros modelos para o entendimento dos resultados; comparar estatísticas entre municípios e utilizar a técnica de cross-validation na etapa de treinamento do modelo.

[43] http://www.nois.ind.puc-rio.br/

# REFERÊNCIAS BIBLIOGRÁFICAS

Abadi, Martín; Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Rafal Jozefowicz, Yangqing Jia, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dan Mané, Mike Schuster, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. Software available from tensorflow.org

Allaire, J. (2012). RStudio: integrated development environment for R. Boston, MA, 770(394), 165-171.

Awais, M., Hassan, SU. & Ahmed, A. Leveraging big data for politics: predicting general election of Pakistan using a novel rigged model. J Ambient Intell Human Comput 12, 4305–4313 (2021). https://doi.org/10.1007/s12652-019-01378-z

Azevedo, Cristina. Grand Challenges Icoda Covid-19 Data Science seleciona três projetos da Fiocruz. 2021. Disponível em: https://portal.fiocruz.br/noticia/grand-challenges-icoda-covid-19-data-science-seleciona-tres-projetos-da-fiocruz

Barreto-Filho, José Augusto Soares; Veiga, André; Correia, Luis Claudio. COVID-19 e Incertezas: Lições do Frontline para a Promoção da Decisão Compartilhada. 2020. Disponível em: https://www.scielo.br/j/abc/a/PbmrwLNszYDYzDsCmGmWThx/?lang=pt. Acesso em: 11 dez. 2021

Baxter, M. (1990). Generalised linear models , by P. McCullagh and JA Nelder. Pp 511.£ 30. 1989. ISBN 0-412-31760-5 (Chapman and Hall). The Mathematical Gazette, 74(469), 320-321.

Braghetto, K., Cordeiro, D. (2014). Introdução à Modelagem e Execução de Workflows Científicos.

Chamberlin, D. D., Astrahan, M. M., Blasgen, M. W., Gray, J. N., King, W. F., Lindsay, B. G., ... & Yost, R. A. (1981). A history and evaluation of System R. Communications of the ACM, 24(10), 632-646.

Chatterjee, Ayan; Gerdes, Martin W.; Martinez, Santiago G.. Statistical Explorations and Univariate Timeseries Analysis on COVID-19 Datasets to Understand the Trend of Disease Spreading and Death. p. 1-27, maio 2020.

Chauhan, N. e Gautam, N. (2015) 'Parametric Comparison of Data Mining Tools ', v, pp. 291 - 298.

Croda, Julio Henrique Rosa; Garcia, Leila Posenato. Resposta imediata da Vigilância em Saúde à epidemia da COVID-19, 2020.

d'Avila, Cristiane. Uma breve história das campanhas de imunização no Brasil: a vacina como direito e cultura 4 nov. 2019. Disponível em: https://www.cafehistoria.com.br/historia-da-vacinacao-no-brasil/. Acesso em: 15 nov. 2021.

Daphne, Ezer; Whitaker, Kirstie. Data science for the scientific life cycle. eLife, 2019, 8.

de Brum Piana, C. F., de Almeida Machado, A., & Selau, L. P. R. (2009). Estatística básica.

de Souza, Guilherme Faveret Garcia; Aguilar, Soraida; Baião, Fernanda Araújo; Simões, Paulo Henrique; Maçaira, Paula Medina. Estudo de Similaridade da Evolução da COVID-19 nos Estados Brasileiros. LIII Simpósio Brasileiro de Pesquisa Operacional, 2021.

E. Deelman, K. Vahi, G. Juve, M. Rynge, S. Callaghan, P. J. Maechling, R. Mayani, W. Chen, R. Ferreira da Silva, M. Livny, and K. Wenger, "Pegasus: a Workflow Management System for Science Automation," Future Generation Computer Systems, vol. 46, p. 17–35, 2015.

G1 - São Paulo. Mapa da vacinação contra Covid-19 no Brasil. 2021. Disponível em: https://especiais.g1.globo.com/bemestar/vacina/2021/mapa-brasil-vacina-covid/. Acesso em: 30 out. 2021.

G1. Vacinação contra a Covid: Brasil passa EUA em taxa de totalmente imunizados. 2021. Disponível em: https://g1.globo.com/saude/coronavirus/vacinas/noticia/2021/11/17/vacinacao-contra-a-covid -brasil-passa-eua-em-taxa-de-totalmente-imunizados.ghtml. Acesso em: 11 dez. 2021.

Golmohammadi, D; Parast, M. M. and Sanders, N. "The Impact of Service Failures on Firm Profitability: Integrating Machine Learning and Statistical Modeling," in IEEE Transactions on Engineering Management, 2020. doi: 10.1109/TEM.2020.3015771.

Guedes, Thaylon; Martins, Lucas Bertelli; Falci, Maria Luiza Furtuozo; Silva, Vitor; Ocaña, Kary A.C.S.; Mattoso, Marta; Bedo, Marcos; de Oliveira, Daniel. Capturing and Analyzing Provenance from Spark-based Scientific Workflows with SAMbA-RaP, Future Generation Computer Systems, Volume 112, 2020, Pages 658-669.

Gujarati, Damodar N., e Down C Porter. 2011. Econometria básica. 5a ed. New York: Mc Graw Hill

História das vacinas: Nerdologia, 2021. Son., P&B. Disponível em: https://www.youtube.com/watch?v=ENttrlq3zmg. Acesso em: 16 nov. 2021.

http://portalsinan.saude.gov.br/calendario-epidemiologico-2020/43-institucional/171-calendario-epidemiologico-2021

https://jupyter-notebook-beginner-guide.readthedocs.io/en/latest/what_is_jupyter.html - Acessado em 05/11/2021 - 23:00

Hunter, J. D. (2007). Matplotlib: A 2D graphics environment. Computing in science & engineering, 9(03), 90-95.

ISTOÉ. Brasil aplica a primeira vacina contra a covid-19 após aprovação da Anvisa. 2021. Disponível em: https://www.istoedinheiro.com.br/brasil-aplica-a-primeira-vacina-contra-a-covid-19-apos-aprovacao-da-anvisa/. Acesso em: 10 nov. 2021.

King, J., & Magoulas, R. (2015). 2015 data science salary survey. O'Reilly Media, Incorporated.

Landeiro, V. L. (2011). Introdução ao uso do programa R. Manaus: Instituto Nacional de Pesquisas da Amazônia.

Lorica, Ben; Armbrust, Michael; Ghodsi, Ali; Xin, Reynold e Zaharia, Matei. O que é um Lakehouse? (2020). https://databricks.com/blog/2020/01/30/what-is-a-data-lakehouse.html?itm_data=lakehouse-link-lakehouseblog

Ludäscher, Bertram; Weske, Mathias; Mcphillips, Timothy; Bowers, Shawn. Scientific Workflows: business as usual?, Berlin, Heidelberg. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009. p. 31-47.

Martinez, I., Viles, E., & Olaizola, I. G. (2021). Data Science Methodologies: Current Challenges and Future Approaches. Big Data Research, 24, 100183.

Mattos, Amanda; Silva, Fabio Coutinho da; Ruberg, Nicolaas; Cruz, Sérgio Manuel Serra da. Gerência de Workflows Científicos: Uma Análise Crítica No Contexto da Bioinformática. Universidade Federal do Rio de Janeiro, Rio de Janeiro, 2008.

Melton, J., & Simon, A. R. (2001). SQL: 1999: understanding relational language components.

Moeschlin, Fredrik. Excel for Data Science? 2018. Disponível em: https://towardsdatascience.com/excel-for-data-science-a82247670d7a

Ninomiya, Vitor Yukio. Vacinação Covid-19: Janssen (Johnson & Johnson). 2021. Disponível em: https://coronavirus.saude.mg.gov.br/blog/331-vacinacao-covid-19-janssen. Acesso em: 10 nov. 2021.

Noronha, T. A importância da vacina no combate ao novo coronavírus. 2021. Disponível em: https://www.uff.br/?q=importancia-da-vacina-no-combate-ao-novo-coronavirus. Acesso em: 16 nov. 2021.

Organization, World Health. Vacina da Oxford/AstraZeneca contra a COVID-19: o que precisa de saber. 2021. Disponível em: https://www.who.int/pt/news-room/feature-stories/detail/the-oxford-astrazeneca-covid-19-vaccine-what-you-need-to-know?gclid=CjwKCAiA7dKMBhBCEiwAO_crFLkGWqV0BZDTxNgAfVpkvAulOG7t1jI7Vwf47fEyCt_G9i98QH9LGxoCd9UQAvD_BwE. Acesso em: 10 nov. 2021.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... & Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. the Journal of machine Learning research, 12, 2825-2830.

Python Software Foundation. Referência da linguagem Python, versão 3.10. Disponível em http://www.python.org

R Core Team (2021). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL https://www.R-project.org/.

R Development Core Team. R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna, Austria, 2008. ISBN 3-900051-07-0.

Reback, Jeff; McKinney, Wes; jbrockmendel, Joris Van den Bossche; Augspurger, Tom; Cloud, Phillip; Sinhrks, gfyoung; Klein, Adam; Roeschke, Matthew; Hawkins, Simon; Tratner, Jeff; She, Chang; Ayd, William; Petersen, Terji; Garcia, Marc; Schendel, Jeremy; Hayden, Andy; Mehyar, Mortada. (2021). pandas-dev / pandas: Pandas 1.3.4 (v1.3.4). Zenodo. https://doi.org/10.5281/zenodo.3715232

Reis, M. M. (2008). Estatística aplicada à Administração. Florianópolis: Departamento de Ciências da Administração/UFSC.

S. Amin, M. I. Uddin, H. H. Al-Baity, M. A. Zeb and M. A. Khan, "Machine learning approach for covid-19 detection on twitter," Computers, Materials & Continua, vol. 68, no.2, pp. 2231–2247, 2021.

Seabold, Skipper e Josef Perktold. “Modelos de estatísticas: modelagem econométrica e estatística com python. ” Anais da 9ª Conferência Python in Science. 2010

Sebesta, R. W. (2018). Conceitos de Linguagens de Programação-11. Bookman Editora.

Setzer, V. W. (1999). Dado, informação, conhecimento e competência. DataGramaZero Revista de Ciência da Informação, n. 0, 28.

Shcherbakov, Maxim & Shcherbakova, Nataliya & Brebels, Adriaan & Janovsky, Timur & Kamaev, Valery. (2014). Lean Data Science Research Life Cycle: A Concept for Data Analysis Software Development. Communications in Computer and Information Science. 466. 708-716. 10.1007/978-3-319-11854-3_61.

Silva, Laryssa Aparecida Machado da. Workflows Científicos - Curso de Ciência da Computação, Universidade Federal de Juiz de Fora, Juiz de Fora, 2007

Stodden, Victoria. The data science life cycle. Communications Of The Acm, [S.L.], v. 63, n. 7, p. 58-66, 18 jun. 2020. Association for Computing Machinery (ACM). http://dx.doi.org/10.1145/3360646.

Theóphilo, C. R. (1998). Algumas reflexões sobre pesquisas empírica em contabilidade. Caderno de estudos, 01-08.

UNA-SUS. Coronavírus: Brasil confirma primeiro caso da doença. 2020. Disponível em: https://www.unasus.gov.br/noticia/coronavirus-brasil-confirma-primeiro-caso-da-doenca. Acesso em: 09 dez. 2021

VanderPlas, J. (2016). Python data science handbook: Essential tools for working with data. " O'Reilly Media, Inc.".

Vieira, Gabriel S.; NASCIMENTO, Mateus I. do. MODELOS DE APRENDIZADO DE MÁQUINA EM SISTEMAS DE WORKFLOWS CIENTÍFICOS. Universidade Federal do Rio de Janeiro, Rio de Janeiro, 2019.

Wickham, Hadley; ggplot2; editora Springer; 2009.

Witten, Ian & Hall, Mark & Frank, Eibe & Holmes, Geoffrey & Pfahringer, Bernhard & Reutemann, Peter. (2009). The WEKA data mining software: An update. SIGKDD Explorations.

Zhao, X., Keikhosrokiani, P. (2022). Sales Prediction and Product Recommendation Model Through User Behavior Analytics. CMC-Computers, Materials & Continua, 70(2), 3855–3874.

# ANEXO I

| Ordem | Campo | Descrição | Categoria |
|---|---|---|---|
| 1 | document_id | Identificador do documento | |
| 2 | paciente_id | Identificador do vacinado | |
| 3 | paciente_idade | Idade do vacinado | |
| 4 | paciente_dataNascimento | Data de nascimento do vacinado | |
| 5 | paciente_enumSexoBiologico | Sexo do vacinado | M = Masculino, F = Feminino |
| 6 | paciente_racaCor_codigo | Código da raça/cor do vacinado | 1; 2; 3; 4; 99 |
| 7 | paciente_racaCor_valor | Descrição da raça/cor do vacinado | 1 = Branca; 2 = Preta; 3 = Parda; 4 = Amarela; 99 = Sem informação |
| 8 | paciente_endereco_coIbgeMunicipio | Código IBGE do município de endereço do vacinado | |
| 9 | paciente_endereco_coPais | Código do país de endereço do vacinado | |
| 10 | paciente_endereco_nmMunicipio | Nome do município de endereço do vacinado | |
| 11 | paciente_endereco_nmPais | Nome do país de endereço do vacinado | |
| 12 | paciente_endereco_uf | Sigla da UF de endereço do vacinado | |
| 13 | paciente_endereco_cep | 5 dígitos para anonimizado e 7 dígitos para identificado | |
| 14 | paciente_nacionalidade_enumNacionalidade | Nacionalidade do vacinado | |
| 15 | estabelecimento_valor | Código do CNES do estabelecimento que realizou a vacinação | |
| 16 | estabelecimento_razaoSocial | Nome/Razão Social do estabelecimento | |
| 17 | estabelecimento_noFantasia | Nome fantasia do estabelecimento | |
| 18 | estabelecimento_municipio_codigo | Código do município do estabelecimento | |
| 19 | estabelecimento_municipio_nome | Nome do município do estabelecimento | |
| 20 | estabelecimento_uf | Sigla da UF do estabelecimento | |
| 21 | vacina_grupo_atendimento_code | Código do grupo de atendimento ao qual pertence o vacinado | |
| 22 | vacina_grupo_atendimento_nome | Nome do grupo de atendimento ao qual pertence o vacinado | |
| 23 | vacina_categoria_code | Código da categoria | |
| 24 | vacina_categoria_nome | Descrição da categoria | |
| 25 | vacina_lote | Número do lote da vacina | |
| 26 | vacina_fabricante_nome | Nome do fabricante/fornecedor | |
| 27 | vacina_fabricante_referencia | CNPJ do fabricante/fornecedor | |
| 28 | vacina_dataAplicacao | Data de aplicação da vacina | |
| 29 | vacina_descricao_dose | Descrição da dose | |
| 30 | vacina_codigo | Código da vacina | |
| 31 | vacina_nome | Nome da vacina/produto | |
| 32 | sistema origem | Nome do sistema de origem | |

Tabela 1: Dicionário de Dados da Campanha da Vacinação contra COVID-19

Fonte: SIPNI (2020)

# ANEXO II

| Coluna | Descrição |
|---|---|
| epidemiological_week | Número da semana epidemiológica. |
| date | Data de coleta dos dados no formato YYYY-MM-DD. |
| order_for_place | Número que identifica a ordem do registro para este local. O registro referente ao primeiro boletim em que esse local aparecer será contabilizado como 1 e os demais boletins incrementarão esse valor. |
| state | Sigla da unidade federativa, exemplo: SP. |
| city | Nome do município (pode estar em branco quando o registro é referente ao Estado, ou ser preenchido com "Importados/Indefinidos"). |
| city_ibge_code | Código IBGE do local. |
| place_type | Tipo de local que esse registro descreve, pode ser city ou state. |
| last_available_date | Data da qual o dado se refere. |
| last_available_confirmed | Número de casos confirmados do último dia disponível igual ou anterior à data date. |
| last_available_confirmed_per_100k_inhabitants | Número de casos confirmados por 100.000 habitantes do último dia disponível igual ou anterior à data date. |
| new_confirmed | Número de novos casos confirmados desde o último dia (pode ser negativo caso a SES remaneje os casos desse município para outro). |
| last_available_deaths | Número de mortes do último dia disponível igual ou anterior à data date. |
| new_deaths | Número de novos óbitos desde o último dia (pode ser negativo caso a SES remaneje os casos desse município para outro). |
| last_available_death_rate | Taxa de mortalidade (mortes / confirmados) do último dia disponível igual ou anterior à data date. |
| estimated_population | População estimada para o local em 2020, segundo o IBGE. |
| estimated_population_2019 | População estimada para esse município/estado em 2019, segundo o IBGE. |
| is_last | Campo pré-computado que diz se esse registro é o mais novo para esse local, pode ser True ou False. |
| is_repeated | Campo pré-computado que diz se as informações nesse registro foram publicadas pela Secretaria Estadual de Saúde no dia date ou se o dado é repetido do último dia em que o dado está disponível (igual ou anterior a date). |

Tabela 2: Dicionário de Dados de casos e óbitos por COVID-19

Fonte: Brasil.IO (2020)

# ANEXO III

A seguir será apresentada a consulta de agrupamento dos dados do SIPNI que foi realizada no PostgreSQL, com o objetivo de reduzir a dimensionalidade desses dados.

```
CREATE TABLE SIPNI_TRATADA_SP as
(SELECT
      cast (estabelecimento_municipio_codigo as int),
      estabelecimento_municipio_nome,
      estabelecimento_uf,
      cast(sem_epidem_ano as int),
      paciente_enumsexobiologico,
      paciente_racacor_valor,
      vacina_categoria_nome,
      vacina_grupoatendimento_nome,
      vacina_descricao_dose,
      vacina_nome,
      sum (qt_vacinados) as qt_vacinados
      FROM
            (SELECT
                  estabelecimento_municipio_codigo,
                  estabelecimento_municipio_nome,
                  estabelecimento_uf,
                  S."sem_epidem_ano",
                  paciente_enumsexobiologico,
                  paciente_racacor_valor,
                  vacina_categoria_nome,
                  vacina_grupoatendimento_nome,
                  CASE  WHEN vacina_codigo = '88'                              THEN 'unica'
                        WHEN vacina_descricao_dose LIKE '1ª%'
                         and vacina_descricao_dose NOT LIKE '%Revacinação%'    THEN 'D1'
                        WHEN vacina_descricao_dose LIKE '2ª%'                  THEN 'D2'
                        ELSE                                                        'D3_reforco'
                  END vacina_descricao_dose,
                  CASE  WHEN vacina_codigo IN ('89', '85')  THEN 'astrazeneca'
```

```
                        WHEN vacina_codigo = '88'           THEN 'janssen'
                        WHEN vacina_codigo = '86'           THEN 'coronavac'
                        WHEN vacina_codigo = '87'           THEN 'pfizer'
                        ELSE 'sem_identificacao'
                  END AS vacina_nome,
                  COUNT (paciente_id) as qt_vacinados
            FROM public."SIPNI_SP"
                  INNER JOIN public."SEM_EPIDEM_2021" as S
                  ON CAST(vacina_dataaplicacao as varchar(10)) = CAST(S."data" as varchar(10))
            WHERE
                  TO_DATE (vacina_dataaplicacao,'YYYY-MM-DD') between TO_DATE('2021-01-17','YYYY-MM-DD')
              and TO_DATE ('2021-09-30','YYYY-MM-DD')
            GROUP BY
                  estabelecimento_municipio_codigo,
                  estabelecimento_municipio_nome,
                  estabelecimento_uf,
                  S."sem_epidem_ano",
                  paciente_enumsexobiologico,
                  paciente_racacor_valor,
                  vacina_categoria_nome,
                  vacina_grupoatendimento_nome,
                  vacina_descricao_dose,
                  vacina_nome,
                  vacina_codigo,
                  vacina_dataaplicacao
            ) as T
GROUP BY
      estabelecimento_municipio_codigo,
      estabelecimento_municipio_nome,
      estabelecimento_uf,
      sem_epidem_ano,
      paciente_enumsexobiologico,
      paciente_racacor_valor,
      vacina_categoria_nome,
      vacina_grupoatendimento_nome,
      vacina_descricao_dose,
```

```
        vacina_nome
)
```