

PONTIFÍCIA UNIVERSIDADE CATÓLICA DE MINAS GERAIS
NÚCLEO DE EDUCAÇÃO A DISTÂNCIA
Pós-graduação *Lato Sensu* em Ciência de Dados e Big Data

Charles Eduardo da Silva Rodrigues Filho

**ABORDAGEM À CRISE DE REFUGIADOS DO SÉCULO XXI E A UTILIZAÇÃO DE
MODELOS DE MACHINE LEARNING PARA O MAPEAMENTO DE ÁREAS DE
CONFLITOS**

Belo Horizonte

2022

Charles Eduardo da Silva Rodrigues Filho

**ABORDAGEM À CRISE DE REFUGIADOS DO SÉCULO XXI E A UTILIZAÇÃO DE
MODELOS DE MACHINE LEARNING PARA O MAPEAMENTO DE ÁREAS DE
CONFLITOS**

Trabalho de Conclusão de Curso apresentado
ao Curso de Especialização em Ciência de
Dados e Big Data como requisito parcial à
obtenção do título de especialista.

Belo Horizonte

2022

SUMÁRIO

| | |
|---|-----------|
| 1. Introdução..... | 5 |
| 1.1. Contextualização..... | 8 |
| 1.2. O problema proposto | 9 |
| 2. Metodologia | 11 |
| 3. Coleta dos dados | 12 |
| 4. Modelagem do banco de dados | 15 |
| 5. Processamento/Tratamento dos dados | 19 |
| 5.1. Tabela países..... | 20 |
| 5.2. Tabela idh..... | 23 |
| 5.3. Tabela desemprego..... | 25 |
| 5.4. Tabela educacao | 26 |
| 5.5. Tabela expectativa_vida | 27 |
| 5.6. Tabela estabilidade_politica..... | 28 |
| 5.7. Tabela controle_corrupcao | 30 |
| 5.8. Tabela efetividade_governo | 31 |
| 5.9. Tabela divisao_etnica | 32 |
| 5.10. Tabela regime_politico..... | 33 |
| 5.11. Tabela terrorismo | 35 |
| 5.12. Tabela perfil_paises | 38 |
| 6. Análise e exploração dos dados..... | 46 |
| 6.1. Exploração superficial sobre a quantidade de territórios estudados..... | 47 |
| 6.2. Exploração específica sobre os territórios estudados | 48 |
| 6.3. Exploração dos registros de ataques terroristas | 55 |
| 7. Criação de modelos de Machine Learning..... | 78 |
| 7.1. Criação de modelos de Machine Learning com a base de dados de ataques terroristas..... | 80 |
| 7.2. Criação de modelos de Machine Learning com a base de indicadores políticos e sociais em conjunto com registros de ataques terroristas | 90 |
| 8. Apresentação dos resultados | 98 |
| 8.1. Apresentação dos resultados da base de ataques terroristas..... | 98 |

| | |
|---|------------|
| 8.2. Apresentação dos resultados da base de indicadores políticos e sociais em conjunto com registros de ataques terroristas | 100 |
| 9. Conclusão | 102 |
| 10. Links | 103 |
| Referências | 103 |
| Links datasets | 106 |

1. Introdução

Migração é um fenômeno que pode ser definido como “*o deslocamento de indivíduos dentro de um espaço geográfico, de forma temporária ou permanente*”.

Desde os primórdios da humanidade o deslocamento de indivíduos dentro de um espaço geográfico foi um dos métodos utilizados pelo homem para encontrar recursos suficientes para garantir a sua sobrevivência. Mudanças climáticas, disponibilidade de alimentos para a comunidade e fatores ambientais eram determinantes para indicar a necessidade de uma nova mobilização.

Durante o desenvolvimento das civilizações as condições relacionadas ao ambiente foram extremamente relevantes para que o homem sentisse a necessidade de se deslocar e alocar-se em outras regiões. Porém, ao longo da história, diversas outras razões como guerras, escravidão e perseguições sujeitaram populações a ondas de exílio, obrigando comunidades inteiras a se deslocarem de forma involuntária.

De acordo com um estudo desenvolvido pela Organização das Nações Unidas - ONU, atualmente, a estimativa é de que 281 milhões de pessoas, aproximadamente 3,6% da população mundial, vivem fora do seu país de origem. Nos dias atuais, os fluxos migratórios podem ser encorajados por diferentes motivações: sociais, políticas, econômicas, naturais, religiosas e culturais; além da caracterização dos diferentes graus de coação.

Migrar, de forma voluntária ou não, não está restrito apenas ao deslocamento entre países ou continentes, mas também na mesma fronteira, ocupando diferentes regiões do mesmo território. Os registros de pessoas internamente deslocadas têm aumentado em comparação a décadas passadas, é o que informa o relatório do Centro de Monitoramento de Deslocamento Interno - o IDMC. Segundo o IDMC, no ano de 2021 aproximadamente 60 milhões de pessoas estavam internamente deslocadas. Ainda de acordo com o IDMC, as principais razões para números elevados são resultantes de conflitos e desastres decorrentes de fenômenos naturais.

No século XXI, crises humanitárias causadas pela intensificação de conflitos, em especial nos continentes asiático e africano, levaram a uma crise migratória conhecida como ‘A crise dos refugiados’. Este termo foi utilizado para denominar o fluxo migratório acentuado de pessoas a outras regiões, sobretudo próximas ao Leste Europeu e ao Sudoeste Asiático. Segundo o Alto-comissariado das Nações Unidas

para Refugiados - ACNUR, na década passada 84 milhões de pessoas foram forçadas a deixar suas casas, e entre elas 26.6 milhões são consideradas refugiadas.

A crise dos refugiados teve o seu auge no ano de 2015, quatro anos após o início da guerra civil na Síria, país que registra o maior número de refugiados no mundo. De acordo com o ACNUR, mais de 25% da população mundial de refugiados vem da Síria, e até o ano de 2021, cerca de 6.7 milhões de sírios procuraram por refúgio em países próximos, como Turquia, Líbano e Jordânia. Além disso, perante o intenso conflito no país, o número registrado de deslocados internos na Síria alcança 6.6 milhões de pessoas, forçando parte da população a se restabelecer em novos locais, sendo que 2.8 milhões de pessoas ainda permanecem em áreas de difícil acesso.

Dados publicados pelo ACNUR em 2021 mostram que 68% da população de refugiados no mundo vem de apenas cinco países: Síria, Venezuela, Afeganistão, Sudão do Sul e Mianmar. Assim como na Síria, grande parte dos refugiados do Afeganistão e Sudão do Sul estão deslocados devido a intensos conflitos nas regiões. Em Mianmar, porém, uma perseguição étnica ao povo Rohingya fez com que uma grande quantidade de pessoas migrasse para a cidade de Bazar de Cox, em Bangladesh, povoando o maior campo de refugiados do mundo atualmente.

Em contrapartida, 39% dos refugiados estão alocados em cinco destinos: Turquia, Colômbia, Uganda, Paquistão e Alemanha. Em virtude da dificuldade em transitar dentro do próprio território devido aos constantes conflitos e perseguições, muitos refugiados optam por se deslocar para países vizinhos. Essa pode ser a razão pela qual quase 85% dos deslocados estão abrigados em países em desenvolvimento, conforme o ACNUR.

No Brasil, três grandes ondas migratórias foram sentidas na última década: em 2010, devido a uma instabilidade política e econômica em seu país de origem, muitos haitianos migraram para a América do Sul, em especial para o Brasil; em 2015, em virtude de uma guerra civil, o Brasil recebeu deslocados da Síria; e mais recentemente em 2018, após uma crise política e social, muitos venezuelanos cruzaram a fronteira adentrando pela região norte do Brasil.

Segundo o Relatório Refúgio em Números de 2020, desenvolvido pelo Ministério da Justiça e Segurança Pública através da análise de dados da Polícia Federal brasileira, do ACNUR e do Comitê Nacional para os Refugiados – Conare publicado em 2021, o Brasil recebeu 28.899 solicitações de reconhecimento da

condição de refugiado, um número bem abaixo do ano anterior, quando o país registrou mais de 82.000 solicitações. De acordo com o Relatório publicado no ano anterior, em 2020 (com dados apresentados até o ano de 2019), cerca de 60% das solicitações de reconhecimento da condição de refugiado foram de deslocados venezuelanos, enquanto quase 23% foram de deslocados haitianos.

É importante, além de destacar toda a diversidade cultural dos países de origem de pessoas deslocadas, que haja também o entendimento das condições que caracterizam os deslocados. Desta forma, através de uma tabela comparativa busca-se explicar com naturalidade o que difere cada termo empregado.

| Termo | Definição |
|---------------------|--|
| Refugiado | Refugiado é alguém que foi forçado a deixar o seu país devido a guerras, violência ou perseguição, muitas vezes sem aviso. Refugiados são impossibilitados de retornar a seu lar a menos que as condições em sua terra nativa estejam seguras. |
| Requerente de asilo | Um requerente de asilo é alguém que também procura por proteção internacional devido a perigos em seu território de origem, porém seu requerimento de status de refugiado ainda não foi determinado legalmente. * |
| Imigrante | Imigrante é o termo utilizado para pessoas que tomam a decisão voluntária de se mudar de seu país de residência com o objetivo de se restabelecer em outro. |
| Migrante | Migrante pode ser caracterizado como qualquer pessoa |

| | |
|--|--|
| | que muda de um local para outro (dentro de seu próprio território ou não) de forma voluntária, geralmente por razões econômicas. |
|--|--|

**O reconhecimento da condição de refugiado é determinado quando o requerente é capaz de provar diante das autoridades que satisfaz corretamente os critérios requeridos para obter a proteção de refugiado.*

Esse trabalho tem como principal objetivo analisar informações culturais, políticas e sociais dos países asiáticos e africanos que atualmente possuem grande parte dos registros de pessoas deslocadas mundialmente. Através da investigação dessas informações busca-se relacionar estatisticamente ocorrências de conflitos internos e indicadores políticos e sociais de países com altos índices de deslocados na última década. A partir dessa análise desenvolver modelos de *machine learning* que identifiquem com alto poder de precisão áreas de risco e sujeitas a ataques e conflitos na Ásia e na África.

1.1. Contextualização

O tema da pesquisa é a relação entre áreas de conflitos e o acentuado fluxo migratório que ocorreu na década de 2010. A pesquisa está delimitada ao período que compreende os anos entre 2010 e 2018, devido ao intenso número de deslocados residentes dos continentes asiático e africano que migraram para outras regiões devido a crises humanitárias, políticas e sociais.

A escolha do período estudado busca analisar o que foi considerada a crise dos refugiados do século XXI. Embora esse termo tenha sido atribuído apenas em 2015, a pesquisa abrange um espaço de tempo que se iniciou em 2010, previamente ao início da guerra civil síria, até o ano de 2018, posteriormente a crise e perseguição ao povo Rohingya em Mianmar.

Através de uma pesquisa realizada analisando dados de deslocamento do ACNUR e do IDMC pôde-se definir as regiões a serem investigadas durante o desenvolvimento da pesquisa. O estudo para correlacionar dados de ataques

terroristas a regiões dos continentes asiático e africano foi desenvolvido por meio da exploração das informações disponibilizadas pelo *Global Terrorism Database* – GTD.

1.2. O problema proposto

Em consequência do aumento exponencial de deslocados na última década, busca-se através dessa pesquisa reunir informações políticas, sociais e culturais de países onde o número de deslocados teve grande destaque e relacioná-las estatisticamente a fim de verificar a existência de padrões.

A partir da investigação de estudos anteriores sobre temas semelhantes, busca-se através dessa pesquisa analisar padrões econômicos, políticos e sociais que possam ser determinantes para identificar áreas de conflitos. Para isso, será feita a utilização de modelos de *machine learning* com o objetivo de mapear regiões dos continentes africano e asiático.

Os estudos utilizados como inspiração para esse trabalho estão mencionados na seção Referências.

A migração é um fenômeno muito antigo, e no decorrer da história da humanidade inúmeros registros de intensos fluxos migratórios causaram grandes impactos no desenvolvimento das civilizações. A escolha por deslocar-se para diferentes regiões se dá por inúmeras razões, sendo elas por motivos econômicos, sociais ou políticos. Entretanto, é importante ressaltar que nem sempre há voluntariedade em deslocar-se.

Fluxos migratórios intensos são resultantes de grandes marcos ao longo da história, e em sua maioria, ocorrem por meio de diferentes formas de imposição e motivados por conflitos ou desigualdades sociais. Durante séculos, por exemplo, ao menos 12 milhões de africanos foram enclausurados e trazidos à força para as Américas no que é conhecido como a diáspora africana. Após a Segunda Guerra Mundial milhões de sobreviventes do Holocausto se tornaram deslocados, imigrando para a Europa Ocidental. No ano de 1948, após uma limpeza étnica na região, mais de 700.000 palestinos-árabes foram expulsos de suas casas na data que ficou conhecida como “Al-Nakba”.

Atualmente, crises humanitárias ainda são assuntos discutidos frequentemente em nosso cotidiano. Na Síria, uma guerra civil que perdura por mais de 10 anos já deixou mais de 500 mil mortos. Em Mianmar, uma limpeza étnica atingindo a minoria

muçulmana Rohingya fez com que milhares migrassem para um campo de concentração em Bangladesh, deixando um país que atualmente passa por um caos político após um golpe militar. Na América do Sul, na tentativa de escapar de uma crise política, da violência e da falta de serviços essenciais, mais de 5 milhões de venezuelanos estão atualmente vivendo ao redor do mundo.

Ataques com motivações ideológicas, sejam elas políticas, sociais, étnicas ou religiosas com objetivos de atingir a população são considerados ataques terroristas. Embora sejam diferenciados de guerras civis ou batalhas devido a sua esporadicidade e durabilidade, afetam diretamente a sociedade. Populações de países que já enfrentam conflitos armados frequentemente são alvos de constantes ataques dessa proporção. De acordo com o *Global Terrorism Database* - GTD, no ano de 2016 pouco mais de 34.600 pessoas morreram em decorrência de ataques terroristas, sendo que 71% das mortes se concentraram em quatro países: Iraque, Afeganistão, Síria e Somália.

De acordo com o *Global Terrorism Database*, para que um evento seja considerado um ataque terrorista é necessário que cumpra ao menos um dos três requisitos:

- Ter motivações políticas, econômicas, religiosas ou sociais;
- Ter intenção de coagir, intimidar ou transmitir alguma mensagem para um público maior do que as vítimas imediatas;
- Ter sido realizado fora do contexto de guerra, na medida que visa não combatentes, isto é, fora dos parâmetros permitidos pelo direito internacional humanitário.

Segundo o GSDRC, um portal de pesquisa sobre governança, desenvolvimento social, resposta humanitária e conflitos, não há uma explicação exata de porquê alguns países sofrem mais com terrorismo do que outros. Ainda de acordo com o portal, alguns dos motivos podem estar ligados a desigualdade social, especialmente ligadas a grupos culturais diferentes e sociedades mais pobres onde há maior risco de guerras civis e propagação de terrorismo.

Conflitos sociais podem ser resultantes de tensões políticas, culturais, religiosas e também ambientais. Um estudo publicado pelo Instituto Potsdam para Pesquisa de Impacto do Clima analisou conflitos armados entre as décadas de 1980 e 2000 e revela que 23% dos conflitos armados em países com muitas diferenças étnicas coincidem com calamidades climáticas. Recentes análises indicam que

eventos climatológicos já contribuíram para conflitos na Síria, país com um grande índice de divisão étnica, além de Somália e Afeganistão.

De acordo com o relatório Refúgio em Números elaborado pelo OBMigra, a partir dos dados do Conare entre os anos de 2011 e 2020, o Brasil reconheceu 48.142 condições de refúgio através da categoria de fundamentação “Grave e Generalizada Violação dos Direitos Humanos (GGVDH)”, refletindo 93,7% do total de reconhecimentos no período, sendo em sua maioria, deslocados de nacionalidade venezuelana. Outras categorias de fundamentação como “Opinião Política” e “Grupo Social” corresponderam, respectivamente, a 0,5% e 0,4% dos reconhecimentos.

Mediante a análise de informações relacionadas a motivações de deslocamento, pretende-se associar regiões de conflitos e ataques terroristas ao intenso fluxo migratório de deslocados oriundos dos continentes asiático e africano. Para associar e definir as regiões que possuem altos níveis de registros será feita a utilização de modelos de *machine learning* para comparar o melhor resultado a fim de identificar áreas de maior perigo e intensidade.

2. Metodologia

Uma análise prévia foi estabelecida para determinar o período de estudo dessa pesquisa. Após avaliar diferentes fontes de informações e desenvolver uma linha do tempo capaz de relacionar acontecimentos recentes com o período objetivo, pôde-se determinar uma metodologia para cada uma das etapas desenvolvidas. A seguir, são descritos os estágios e procedimentos empregados durante o processo de elaboração do trabalho.

Após a escolha do período a ser estudado foi realizada a avaliação das regiões a serem exploradas durante a pesquisa. Para isso, foram reunidos dados sobre refugiados e deslocados durante o período definido na pesquisa.

Inicialmente foram verificadas informações sobre o número de deslocados no período que compreende o estudo.

```

num_deslocados_acnur_anual = analise_deslocados.groupby(['ano'])['refugiados_acnur'].sum()
print("Quantidade anual de refugiados sob o mandato do ACNUR: " + str(num_deslocados_acnur_anual))

Quantidade anual de refugiados sob o mandato do ACNUR: ano
2010    10548835
2011    10403937
2012    10497017
2013    11698233
2014    14384289
2015    16110276
2016    17184286
2017    19940566
2018    20359553
Name: refugiados_acnur, dtype: int64

```

Em seguida, uma nova análise, dessa vez com a quantidade de deslocados em cada continente entre 2010 e 2018.

```

num_deslocados_acnur = analise_deslocados.groupby([analise_deslocados['continente']])['refugiados_acnur'].sum()
print("Quantidade anual de refugiados sob o mandato do ACNUR por continente: " + str(num_deslocados_acnur))

Quantidade anual de refugiados sob o mandato do ACNUR por continente: continente
Africa     44928461
America   3909372
Asia       77057743
Europe     3588646
Oceania    13751
Name: refugiados_acnur, dtype: int64

```

Após indicação dos altos números de refugiados nos continentes africano e asiático optou-se por desenvolver a pesquisa especialmente sobre as regiões de ambos os continentes.

Durante o desenvolvimento do trabalho diversas ferramentas foram utilizadas com o propósito de agregar maior conhecimento à pesquisa através da avaliação das melhores práticas para cada estágio.

Os dados coletados foram armazenados e tratados em um banco de dados utilizando o gerenciador de servidores web XAMPP em sua versão v3.3.0. A ferramenta utilizada para tratamento dos dados armazenados foi o gerenciador de banco de dados MySQL através da plataforma PHPMyAdmin na versão 5.1.1. Parte dos dados também foram tratados utilizando o Microsoft Excel.

A inserção de informações no banco de dados local foi realizada através do desenvolvimento de scripts utilizando a linguagem de marcação HTML 5 e a linguagem de programação PHP 8.0.10. A conexão com o banco de dados local foi realizada com o servidor web Apache.

Os processos de análise e exploração dos dados e o desenvolvimento dos modelos de *machine learning* foram realizados utilizando a linguagem de programação Python em sua versão 3.8.2 através da versão 6.4.5 do Jupyter Notebook do Anaconda3.

3. Coleta dos dados

Os dados estudados nesta pesquisa foram coletados a partir da análise de informações sistematizadas disponibilizadas por diferentes portais e provedores de dados de caráter público. Todos os dados provêm de plataformas *online* que viabilizam o acesso a dados abertos com o intuito de enriquecer o conhecimento público.

As origens dos dados utilizados no desenvolvimento dessa pesquisa foram determinadas a partir de pesquisas em portais de organizações intergovernamentais, centros de pesquisas sociais e estudos de índices políticos, sociais e culturais. Antes de serem descarregadas houve uma verificação da autenticidade das informações e dos estudos relacionados.

Essa pesquisa busca refletir e alertar sobre o grande fluxo migratório ocorrido na década passada. Para isso, foi utilizada a base de dados do ACNUR como auxiliar para determinar os principais pontos territoriais a serem destacados, levando em consideração os países de origem dos deslocados. Para descarregamento das informações alguns requisitos foram determinados: o período de análise, a caracterização dos deslocados e informações sobre regiões de origem.

As informações apresentando índices sociais referentes ao IDH, percentual de desemprego e vulnerabilidade, índice de educação e expectativa de vida foram recolhidas através da pesquisa no portal de Relatório de Desenvolvimento Humano da ONU. Em sua maioria, os dados são referentes ao período entre 1990 e 2019. Posteriormente, houve um tratamento específico para que os dados se adequassem ao período da pesquisa.

Para a análise dos índices de divisões étnicas foi utilizado o dataset “*The Historical Index of Ethnic Fractionalization (HIEF)* ”. O estudo publicado pelo *Robert Schuman Centre for Advanced Studies* (RSCAS) através do *European University Institute* (EUI) em 2019 sobre o índice de divisão étnica contém informações sobre 165 países ao redor do mundo entre os anos de 1945 e 2013. O resultado deste estudo está disponível em diversas plataformas, como no próprio EUI-RSCAS e no portal de datasets da *Harvard University*.

A pesquisa sobre regimes políticos ao redor do mundo é disponibilizada pelo portal *Our World In Data*, que desenvolveu diversas pesquisas sobre direitos democráticos e regimes políticos ao longo dos anos. A classificação dos regimes políticos utiliza dados do projeto “*Variety of Democracy (V-Dem)* ” que diferencia cada categoria entre quatro tipos de sistemas políticos: *closed autocracy, electoral*

autocracy, electoral democracy e liberal democracy. A classificação é determinada de acordo com alguns critérios. A seguir uma tradução literal e explicativa de cada regime.

| Regime | Tradução | Definição | |
|---------------------|----------------------|---|---|
| closed autocracy | autocracia fechada | Eleições unipartidárias para o chefe executivo ou para o legislativo | Na prática, as eleições não são multipartidárias, ou livres e justas. |
| electoral autocracy | autocracia liberal | Pela lei, eleições multipartidárias para o chefe executivo ou para o legislativo. | |
| electoral democracy | democracia eleitoral | O Estado de direito, ou princípios liberais não exercidos. | Na prática, as eleições são multipartidárias, livres e justas. |
| liberal democracy | democracia liberal | Tanto o Estado de direito quanto princípios liberais são exercidos. | |

O recolhimento dos dados sobre religião é fundamentado no estudo desenvolvido pelo *Pew Research Center*, um *think tank* especializado em tendências políticas e sociais globais. No ano de 2012 pesquisadores analisaram registros demográficos para desenvolver uma projeção sobre a fracionalização de religiões em todos os territórios, até a década de 2050. Com mais de 1.200 registros, os dados recolhidos foram tratados de forma que as religiões predominantes em cada território fossem destacadas durante o desenvolvimento dessa pesquisa.

Os indicadores de controle de corrupção, efetividade do governo e estabilidade política e controle da violência são índices apresentados pelo Banco Mundial através do *Worldwide Governance Indicators* — WGI, um conjunto de pesquisa fornecida por diversas pessoas, empresas e especialistas renomados em países industrializados e em desenvolvimento. O descarregamento dessas informações foi feito diretamente no portal de *downloads* da instituição. Antes de serem inseridas em uma base de dados, as informações foram verificadas e optou-se por utilizar índices percentuais sobre cada um dos indicadores.

O principal objetivo de estudo dessa pesquisa, a base de dados sobre ataques terroristas, é disponibilizada pelo *Global Terrorism Database*. O dataset contém registros de eventos desde o ano de 1970 até 2019 e possui mais de 200.000 linhas e mais de 100 colunas.

Parte da inserção de alguns dados foi feita de forma manual. Uma tabela auxiliar com informações sobre os países e seus códigos padronizados de acordo com a ISO 3166 foi utilizada para identificar registros faltantes na tabela GTD. Utilizando essas informações foi desenvolvido um pequeno formulário em linguagem de marcação HTML e linguagem de programação PHP para realizar uma conexão com o banco de dados e inserir as informações diretamente nele.

Durante o agrupamento das informações e o descarregamento dos arquivos não foi necessário a utilização de ferramentas para extrair informações diretamente da web ou API's.

Todas as origens dos dados estão referenciadas na seção de Referências.

4. Modelagem do banco de dados

Após a coleta de todos os dados utilizados para pesquisa, um banco de dados foi desenvolvido como método para armazenar e manipular as informações reunidas. Dessa maneira, através de um sistema de gerenciamento de banco de dados relacional baseado em linguagem de consulta estruturada foi elaborado o banco de dados.

Para a construção do banco de dados local, o gerenciador de servidores web XAMPP foi utilizado para conexão com o serviço de banco de dados gerenciado MySQL e das aplicações em nuvem necessárias para complementar o recolhimento

e tratamento de dados. Através da plataforma *online* PHPMyAdmin foi possível administrar o banco de dados pela internet.

A primeira etapa para o desenvolvimento foi criar o banco de dados. Na figura abaixo é possível visualizar de forma simplificada o início dos processos de desenvolvimento e gerenciamento do banco e a manipulação dos dados.

```
MySQL não retornou nenhum registo. (A consulta demorou 0,0154 segundos.)  
CREATE DATABASE projeto CHARACTER SET utf8 COLLATE utf8_swedish_ci;  
[ Editar em linha ] [ Edita ] [ Criar código PHP ]
```

Após a criação do banco de dados, as tabelas (relações) foram construídas de acordo com a disposição dos dados reunidos em arquivos em formato de valores separados por vírgulas (CSV). As tabelas foram projetadas dessa forma para garantir, num primeiro momento, a adequação dos registros em linhas (tuplas) e colunas (atributos).

Previamente a criação das tabelas mais importantes para o desenvolvimento da pesquisa, foi construída a relação de dados organizados para identificação e padronização dos países, nomeada *paises*. Como mencionado anteriormente, os dados provenientes dessa relação foram digitados manualmente.

```
MySQL não retornou nenhum registo. (A consulta demorou 0,0141 segundos.)  
CREATE TABLE `paises` ( `id` int(4) NOT NULL, `codigo_continente` int(1) DEFAULT NULL, `nome_continente` varchar(10) COLLATE utf8_swedish_ci DEFAULT NULL, `codigo_regiao_gtd` int(2) DEFAULT NULL, `nome_regiao` varchar(30) COLLATE utf8_swedish_ci DEFAULT NULL, `codigo_pais_gtd` int(4) DEFAULT NULL, `nome_pais` varchar(96) COLLATE utf8_swedish_ci DEFAULT NULL, `iso` varchar(2) COLLATE utf8_swedish_ci DEFAULT NULL );  
[ Editar em linha ] [ Edita ] [ Criar código PHP ]
```

Seguindo o padrão para construção de tabelas, foram criadas a seguir as relações contendo os dados organizados sobre os índices sociais do Relatório de Desenvolvimento Humano da ONU.

Primeiramente, foi criada a tabela para receber os registros dos Índices de Desenvolvimento Humano entre os anos de 1990 e 2019, nomeada *idh*.

```
MySQL não retornou nenhum registo. (A consulta demorou 0,0507 segundos.)  
CREATE TABLE `idh` ( `idh_rank` int(3) DEFAULT NULL, `nome_pais` varchar(96) COLLATE utf8_swedish_ci DEFAULT NULL, `ano_1990` decimal(10,3) COLLATE utf8_swedish_ci DEFAULT NULL, `ano_1991` decimal(10,3) COLLATE utf8_swedish_ci DEFAULT NULL, `ano_1992` decimal(10,3) COLLATE utf8_swedish_ci DEFAULT NULL, `ano_1993` decimal(10,3) COLLATE utf8_swedish_ci DEFAULT NULL, `ano_1994` decimal(10,3) COLLATE utf8_swedish_ci DEFAULT NULL, `ano_1995` decimal(10,3) COLLATE utf8_swedish_ci DEFAULT NULL, `ano_1996` decimal(10,3) COLLATE utf8_swedish_ci DEFAULT NULL, `ano_1997` decimal(10,3) COLLATE utf8_swedish_ci DEFAULT NULL, `ano_1998` decimal(10,3) COLLATE utf8_swedish_ci DEFAULT NULL, `ano_1999` decimal(10,3) COLLATE utf8_swedish_ci DEFAULT NULL, `ano_2000` decimal(10,3) COLLATE utf8_swedish_ci DEFAULT NULL, `ano_2001` decimal(10,3) COLLATE utf8_swedish_ci DEFAULT NULL );  
[ Edita ]
```

A seguir, a tabela que receberá os registros do percentual de desemprego entre os anos de 1991 e 2019 foi criada levando o nome *desemprego*.

```
MySQL não retornou nenhum registo. (A consulta demorou 0,0226 segundos.)  
CREATE TABLE desemprego( idh_rank INT(3), nome_pais VARCHAR(96), ano_1991 decimal(10,1), ano_1995 decimal(10,1), ano_2000 decimal(10,1), ano_2005 decimal(10,1), ano_2010 decimal(10,1), ano_2011 decimal(10,1), ano_2012 decimal(10,1), ano_2013 decimal(10,1), ano_2014 decimal(10,1), ano_2015 decimal(10,1), ano_2016 decimal(10,1), ano_2017 decimal(10,1), ano_2018 decimal(10,1), ano_2019 decimal(10,1) );  
[ Editar em linha ] [ Edita ] [ Criar código PHP ]
```

Seguidamente, a tabela *educação* foi criada para inserção dos registros de índice de educação dos países entre os anos de 1990 a 2019.

```
MySQL não retornou nenhum registo. (A consulta demorou 0,0468 segundos)

CREATE TABLE `educacao` ( `idh_rank` int(3) DEFAULT NULL, `nome_pais` varchar(96) COLLATE utf8_swedish_ci DEFAULT NULL, `ano_1990` decimal(10,3)
COLLATE utf8_swedish_ci DEFAULT NULL, `ano_1991` decimal(10,3) COLLATE utf8_swedish_ci DEFAULT NULL, `ano_1992` decimal(10,3) COLLATE
utf8_swedish_ci DEFAULT NULL, `ano_1993` decimal(10,3) COLLATE utf8_swedish_ci DEFAULT NULL, `ano_1994` decimal(10,3) COLLATE utf8_swedish_ci
DEFAULT NULL, `ano_1995` decimal(10,3) COLLATE utf8_swedish_ci DEFAULT NULL, `ano_1996` decimal(10,3) COLLATE utf8_swedish_ci DEFAULT NULL,
`ano_1997` decimal(10,3) COLLATE utf8_swedish_ci DEFAULT NULL, `ano_1998` decimal(10,3) COLLATE utf8_swedish_ci DEFAULT NULL, `ano_1999` decimal(10,3)
COLLATE utf8_swedish_ci DEFAULT NULL, `ano_2000` decimal(10,3) COLLATE utf8_swedish_ci DEFAULT NULL, `ano_2001` decimal(10,3) COLLATE
utf8_swedish_ci DEFAULT NULL ) ENGINE=InnoDB DEFAULT CHARSET=utf8_swedish_ci;
```

[Edita]

Em seguida, a tabela *expectativa_vida* foi elaborada para receber os dados de expectativa de vida em anos dos territórios entre os anos de 1990 e 2019.

```
MySQL não retornou nenhum registo. (A consulta demorou 0,0405 segundos)

CREATE TABLE `expectativa_vida` ( `idh_rank` int(3) DEFAULT NULL, `nome_pais` varchar(96) COLLATE utf8_swedish_ci DEFAULT NULL, `ano_1990` decimal(10,1)
COLLATE utf8_swedish_ci DEFAULT NULL, `ano_1991` decimal(10,1) COLLATE utf8_swedish_ci DEFAULT NULL, `ano_1992` decimal(10,1)
COLLATE utf8_swedish_ci DEFAULT NULL, `ano_1993` decimal(10,1) COLLATE utf8_swedish_ci DEFAULT NULL, `ano_1994` decimal(10,1) COLLATE
utf8_swedish_ci DEFAULT NULL, `ano_1995` decimal(10,1) COLLATE utf8_swedish_ci DEFAULT NULL, `ano_1996` decimal(10,1) COLLATE utf8_swedish_ci
DEFAULT NULL, `ano_1997` decimal(10,1) COLLATE utf8_swedish_ci DEFAULT NULL, `ano_1998` decimal(10,1) COLLATE utf8_swedish_ci DEFAULT NULL,
`ano_1999` decimal(10,1) COLLATE utf8_swedish_ci DEFAULT NULL, `ano_2000` decimal(10,1) COLLATE utf8_swedish_ci DEFAULT NULL, `ano_2001` decimal(10,1)
COLLATE utf8_swedish_ci DEFAULT NULL ) ENGINE=InnoDB DEFAULT CHARSET=utf8_swedish_ci;
```

[Edita]

Para a criação da relação de dados sobre os regimes políticos dos países houve um tratamento anterior à construção da tabela no banco de dados. Parte dos dados apresentados na tabela de estudo foram removidos previamente no Microsoft Excel após análise através do software *Statistical Package for the Social Sciences - SPSS*, já que não se tinha intenção de utilizá-los nesta pesquisa. A parte que continha dados úteis para o desenvolvimento da pesquisa auxiliou na modelagem da estrutura da relação *regime_politico*.

```
MySQL não retornou nenhum registo. (A consulta demorou 0,0146 segundos)

CREATE TABLE `regime_politico` ( `nome_pais` varchar(96) COLLATE utf8_swedish_ci DEFAULT NULL, `ano` int(4) DEFAULT NULL, `codigo_regime` int(2)
DEFAULT NULL, `descricao_regime` varchar(57) COLLATE utf8_swedish_ci DEFAULT NULL );
```

[Editar em linha] [Edita] [Criar código PHP]

A relação de dados sobre religiões estava dividida em duas partes: o número estimado dos praticantes da religião e a representação desses números em percentuais, porém não houve necessidade de incluí-las no banco de dados, uma vez que foram analisadas através do Microsoft Excel e incluídas manualmente.

Os indicadores de controle de corrupção, efetividade do governo e estabilidade política e controle da violência apresentados pelo Banco Mundial foram descarregados em apenas um arquivo de planilhas com diferentes abas. Antes de serem inseridas no banco de dados em suas respectivas tabelas, as informações foram dispostas em arquivos de valores separados por vírgulas de maneira individual.

Após a separação das informações em arquivos diferentes, as três tabelas passaram por um tratamento antes de serem inseridas no banco de dados. Após analisar as informações optou-se por utilizar os índices percentuais de cada indicador.

Para isso, registros fora do período de estudo e informações desnecessárias foram removidos.

Assim, as tabelas puderam ser modeladas e criadas.

Primeiramente, a tabela *controle_corrupcao* foi criada para inserção dos indicadores de controle de corrupção entre os anos de 2010 e 2018.

```
✓ MySQL não retornou nenhum registo. (A consulta demorou 0,0348 segundos)

CREATE TABLE `controle_corrupcao` ( `nome_pais` varchar(96) COLLATE utf8_swedish_ci DEFAULT NULL, `perc_ano_2010` decimal(10,2) DEFAULT NULL,
`perc_ano_2011` decimal(10,2) DEFAULT NULL, `perc_ano_2012` decimal(10,2) DEFAULT NULL, `perc_ano_2013` decimal(10,2) DEFAULT NULL, `perc_ano_2014` decimal(10,2) DEFAULT NULL, `perc_ano_2015` decimal(10,2) DEFAULT NULL, `perc_ano_2016` decimal(10,2) DEFAULT NULL, `perc_ano_2017` decimal(10,2) DEFAULT NULL, `perc_ano_2018` decimal(10,2) DEFAULT NULL );
```

[Editar em linha] [Edita] [Criar código PHP]

Em seguida, a tabela *efetividade_governo* foi criada para inserção dos indicadores de efetividade do governo entre os anos de 2010 e 2018.

```
✓ MySQL não retornou nenhum registo. (A consulta demorou 0,0150 segundos)

CREATE TABLE `efetividade_governo` ( `nome_pais` varchar(96) COLLATE utf8_swedish_ci DEFAULT NULL, `perc_ano_2010` decimal(10,2) DEFAULT NULL,
`perc_ano_2011` decimal(10,2) DEFAULT NULL, `perc_ano_2012` decimal(10,2) DEFAULT NULL, `perc_ano_2013` decimal(10,2) DEFAULT NULL, `perc_ano_2014` decimal(10,2) DEFAULT NULL, `perc_ano_2015` decimal(10,2) DEFAULT NULL, `perc_ano_2016` decimal(10,2) DEFAULT NULL, `perc_ano_2017` decimal(10,2) DEFAULT NULL, `perc_ano_2018` decimal(10,2) DEFAULT NULL );
```

[Editar em linha] [Edita] [Criar código PHP]

Depois, a tabela *estabilidade_politica* foi criada para inserção dos indicadores de estabilidade política e controle da violência entre os anos de 2010 e 2018.

```
✓ MySQL não retornou nenhum registo. (A consulta demorou 0,0161 segundos)

CREATE TABLE `estabilidade_politica` ( `nome_pais` varchar(96) COLLATE utf8_swedish_ci DEFAULT NULL, `perc_ano_2010` decimal(10,2) DEFAULT NULL,
`perc_ano_2011` decimal(10,2) DEFAULT NULL, `perc_ano_2012` decimal(10,2) DEFAULT NULL, `perc_ano_2013` decimal(10,2) DEFAULT NULL, `perc_ano_2014` decimal(10,2) DEFAULT NULL, `perc_ano_2015` decimal(10,2) DEFAULT NULL, `perc_ano_2016` decimal(10,2) DEFAULT NULL, `perc_ano_2017` decimal(10,2) DEFAULT NULL, `perc_ano_2018` decimal(10,2) DEFAULT NULL );
```

[Editar em linha] [Edita] [Criar código PHP]

Logo após a criação das tabelas de indicadores políticos globais, foi realizada a elaboração da tabela *divisao_etnica*, criada para receber dados referentes ao índice de divisão étnica de 165 países.

```
✓ MySQL não retornou nenhum registo. (A consulta demorou 0,0178 segundos)

CREATE TABLE divisao_etnica( nome_pais VARCHAR(96), ano INT(4), indice DECIMAL(10,3) );
```

[Editar em linha] [Edita] [Criar código PHP]

Para a criação da relação de dados principal que contém registros de eventos terroristas desde a década de 1970, foi necessário realizar uma “limpeza” prévia de alguns atributos diretamente no Microsoft Excel. Algumas colunas foram removidas para que os registros contendo os atributos previsores e o atributo classe fossem diretamente inseridos no banco.

Após a tomada de decisão e a escolha dos atributos, a tabela *terrorismo* foi elaborada.

```

CREATE TABLE `terrorismo` (
  `id_evento` varchar(12) DEFAULT NULL,
  `ano` int(4) DEFAULT NULL,
  `mes` int(2) DEFAULT NULL,
  `dia` int(2) DEFAULT NULL,
  `codigo_pais` int(4) DEFAULT NULL,
  `nome_pais` varchar(96) COLLATE utf8_swedish_ci DEFAULT NULL,
  `codigo_regiao` int(2) DEFAULT NULL,
  `nome_regiao` varchar(30) COLLATE utf8_swedish_ci DEFAULT NULL,
  `provincia` varchar(96) COLLATE utf8_swedish_ci DEFAULT NULL,
  `cidade` varchar(96) COLLATE utf8_swedish_ci DEFAULT NULL,
  `criterio1` int(1) DEFAULT NULL,
  `criterio2` int(1) DEFAULT NULL,
  `criterio3` int(1) DEFAULT NULL,
  `dvida_terrorismo` int(1) DEFAULT NULL,
  `sucesso_ataque` int(1) DEFAULT NULL,
  `codigo_metodo_ataque` int(1) DEFAULT NULL,
  `incidente_alternativo` varchar(30) COLLATE utf8_swedish_ci DEFAULT NULL,
  `ataques_conectados` int(1) DEFAULT NULL,
  `codigo_tipo_vitima` int(2) DEFAULT NULL,
  `tipo_vitima` varchar(35) COLLATE utf8_swedish_ci DEFAULT NULL,
  ...
) [Edita]

```

Inicialmente, o planejamento das relações de dados foi combinar as informações das planilhas descarregadas e inseri-las em tabelas, sem realizar muitas alterações nos dados. Posteriormente, para que os dados pudessem ser tratados, novas configurações foram feitas, como a definição de chaves primárias, alteração e adição de novos campos e a combinação de tabelas.

5. Processamento/Tratamento dos dados

Modelado o banco de dados, deu-se início aos processos de tratamento dos dados e enriquecimento das bases.

Os procedimentos de tratamento dos dados e enriquecimento das bases, embora semelhantes em certos momentos, tornaram-se diferentes a cada novo processamento em razão das peculiaridades encontradas nas informações analisadas.

Por conta dessa dissimilaridade, optou-se por discriminar os processos de cada tabela em seções distintas, definidas por categorias. As categorias estão listadas abaixo e os processos para cada uma das tabelas serão detalhados a seguir.

| Tabela | Categoria |
|------------------|--|
| paises | Formulário Online |
| idh | Relatório de Desenvolvimento Humano da ONU |
| desemprego | Relatório de Desenvolvimento Humano da ONU |
| educacao | Relatório de Desenvolvimento Humano da ONU |
| expectativa_vida | Relatório de Desenvolvimento Humano da ONU |

| | |
|-----------------------|--|
| religiao | Inserção manual no banco de dados |
| controle_corrupcao | Indicadores de Governança Mundial do Banco Mundial |
| efetividade_governo | Indicadores de Governança Mundial do Banco Mundial |
| estabilidade_politica | Indicadores de Governança Mundial do Banco Mundial |
| divisa_etnica | Sem Categoria Específica |
| regime_politico | Sem Categoria Específica |
| terrorismo | Sem Categoria Específica |
| perfil_paises | Concatenação de Tabelas |

5.1. Tabela países

As tarefas de processamento dos dados iniciaram-se pela tabela *paises*. Ao contrário das outras relações, as informações dessa tabela foram inseridas manualmente no banco de dados.

Com a elaboração de um *script* utilizando linguagem de marcação HTML e linguagem de programação PHP um formulário foi criado para que as informações digitadas fossem inseridas diretamente no banco de dados. Parte das informações foram retiradas do livro de metodologias disponibilizado pelo *Global Terrorism Database* e a outra parte (códigos ISO) foram recolhidas de um repositório público de dados.

Nas imagens abaixo é possível verificar as linhas de código utilizadas para elaboração do *script* do formulário para inserção dos dados.

```
73 <div class="row">
74 </div>
75 <div class="tile mb-4">
76 <div class="row">
77 <div class="col-lg-12">
78 <h4 class="line-head">Inserir Descrição Paises</h4>
79 <form method="POST">
80 <div class="row mb-4">
81 <div class="col-md-2">
82 <label>Código Continente</label>
83 <input type="text" name="codigo_continente" class="form-control"
autocomplete="off">
84 </div>
85 <div class="col-md-10">
86 <label>Continente</label>
87 <input type="text" name="nome_continente" class="form-control"
autocomplete="off">
88 </div>
89 </div>
90 <div class="row mb-4">
91 <div class="col-md-2">
92 <label>Código Região</label>
93 <input type="text" name="codigo_regiao" class="form-control"
autocomplete="off">
94 </div>
95 <div class="col-md-10">
96 <label>Nome Região</label>
97 <input type="text" name="nome_regiao" class="form-control" autocomplete=
"off">
98 </div>
99 </div>
100
101 <div class="row mb-4">
102 <div class="col-md-2">
103 <label>Código País</label>
104 <input type="text" name="codigo_pais" class="form-control" autocomplete=
"off">
105 </div>
106 <div class="col-md-9">
107 <label>Nome País</label>
108 <input type="text" name="nome_pais" class="form-control" autocomplete=
"off">
109 </div>
110 <div class="col-md-1">
111 <label>ISO 3166-1</label>
112 <input type="text" name="iso" class="form-control" autocomplete="off">
113 </div>
114 </div>
115 <div class="row mb-12">
116 <div class="col-md-12">
117 <button class="btn btn-primary" type="submit" style=" float:right"><i
class="fa fa-fw fa-lg fa-check-circle"></i> Enviar</button>
118 </div>
119 </div>
120 </form>
```

```

178 <?php
179
180 //ini_set('default_charset','UTF-8');
181 $study = mysqli_connect ("localhost", "root", "", "projeto");
182 $study->set_charset('utf8');
183
184
185 if (isset($_POST['codigo_regiao'])) // verifica se as variáveis existem
186 {
187     if ($_POST['codigo_regiao'] != NULL)
188     {
189
190     $codigo_continente = $_POST["codigo_continente"];
191     $nome_continente = $_POST["nome_continente"];
192     $codigo_regiao = $_POST["codigo_regiao"];
193     $nome_regiao = $_POST["nome_regiao"];
194     $codigo_pais = $_POST["codigo_pais"];
195     $nome_pais = $_POST["nome_pais"];
196     $iso = $_POST["iso"];
197
198     $sql = "INSERT INTO paises VALUES";
199     $sql .= "('{$codigo_continente}', '{$nome_continente}', '{$codigo_regiao}', '{$nome_regiao}', '$codigo_pais', '{$nome_pais}', '{$iso}')";
200     $resultado = mysqli_query ($study, $sql);
201     mysqli_close($study);
202
203
204 echo "<script language='JavaScript'>
205 if($resultado){
206         swal('Sucesso!', 'Novo item cadastrado', 'success');
207         setTimeout(function () {
208             window.location.href = 'cadastro.php';
209         }, 1000)
210     }
211 else{
212     swal('Sucesso!', 'Não foi possível realizar o cadastro.', 'error');
213 }
214
215 </script>";
216     }
217 }
218 echo "<script>
219     if ( window.history.replaceState ) {
220         window.history.replaceState( null, null, window.location.href );
221     }
222 </script>";
223 ?>
```

Inserir Descrição Países

| | | |
|---------------------------------------|----------------------|----------------------|
| Código Continente | Continente | |
| <input type="text"/> | <input type="text"/> | |
| Código Região | Nome Região | |
| <input type="text"/> | <input type="text"/> | |
| Código País | Nome País | ISO 3166-1 |
| <input type="text"/> | <input type="text"/> | <input type="text"/> |
| <input type="button" value="Enviar"/> | | |

Através do comando *describe* do MySQL é possível ver as informações da tabela, como nome dos campos, tipos e relações.

| Field | Type | Null | Key | Default | Extra |
|-------------------|-------------|------|-----|---------|----------------|
| id | int(4) | NO | PRI | NULL | auto_increment |
| codigo_continente | int(1) | NO | | NULL | |
| nome_continente | varchar(10) | NO | | NULL | |
| codigo_regiao_gtd | int(2) | YES | | NULL | |
| nome_regiao | varchar(30) | YES | | NULL | |
| codigo_pais_gtd | int(4) | YES | | NULL | |
| nome_pais | varchar(96) | YES | | NULL | |
| iso | varchar(2) | YES | | NULL | |

A seguir, as tabelas com informações do Relatório de Desenvolvimento Humano da ONU foram as próximas a serem processadas.

De maneira muito semelhante uma das outras, todas passaram pelos mesmos processos de tratamento dos dados. Por conta dessa semelhança, todos os processos serão descritos, porém apenas a primeira tabela referenciada conterá imagens, já que os processos foram os mesmos para todas.

5.2. Tabela idh

Iniciando pela tabela *idh*, as informações foram inseridas de acordo com a disposição dos registros no arquivo CSV.

```
✓ 189 linha(s) inseridas. (A consulta demorou 0,0128 segundos.)
LOAD DATA INFILE "C:/Users/Spaox/Documents/Projeto/Arquivos/Tabelas/Tratadas/idh.csv" INTO TABLE idh character set latin1 FIELDS TERMINATED BY ",";
[ Editar em linha ] [ Edita ] [ Criar código PHP ]
```

Em seguida, duas novas colunas foram adicionadas: as colunas *id* e *iso*. As colunas foram adicionadas com a intenção de criar identificações para cada registro e território.

```
✓ MySQL não retornou nenhum registo. (A consulta demorou 0,0420 segundos.)
ALTER TABLE idh ADD COLUMN( id INT(3) AUTO_INCREMENT PRIMARY KEY, iso VARCHAR(2) );
[ Editar em linha ] [ Edita ] [ Criar código PHP ]
```

Logo após a criação das duas novas colunas, uma função *join* foi utilizada para atualizar os campos *iso* com dados da tabela *paises* utilizando o nome dos territórios como referência.

```
✓ 163 linha(s) afectadas. (A consulta demorou 0,0479 segundos.)
UPDATE idh INNER JOIN paises ON idh.nome_pais = paises.nome_pais SET idh.iso = paises.iso;
[ Editar em linha ] [ Edita ] [ Criar código PHP ]
```

Os registros que não foram atualizados após a função *join* foram atualizados manualmente.

Depois da atualização dos campos *iso*, registros com valores “0.0” foram atualizados para “NULL”, já que registros nulos foram inseridos com valores “0.0” na tabela.

```

1 UPDATE `idh` SET `ano_1990`=NULL WHERE `ano_1990`=0.0;
2 UPDATE `idh` SET `ano_1991`=NULL WHERE `ano_1991`=0.0;
3 UPDATE `idh` SET `ano_1992`=NULL WHERE `ano_1992`=0.0;
4 UPDATE `idh` SET `ano_1993`=NULL WHERE `ano_1993`=0.0;
5 UPDATE `idh` SET `ano_1994`=NULL WHERE `ano_1994`=0.0;
6 UPDATE `idh` SET `ano_1995`=NULL WHERE `ano_1995`=0.0;
7 UPDATE `idh` SET `ano_1996`=NULL WHERE `ano_1996`=0.0;
8 UPDATE `idh` SET `ano_1997`=NULL WHERE `ano_1997`=0.0;
9 UPDATE `idh` SET `ano_1998`=NULL WHERE `ano_1998`=0.0;
10 UPDATE `idh` SET `ano_1999`=NULL WHERE `ano_1999`=0.0;
11 UPDATE `idh` SET `ano_2000`=NULL WHERE `ano_2000`=0.0;
12 UPDATE `idh` SET `ano_2001`=NULL WHERE `ano_2001`=0.0;
13 UPDATE `idh` SET `ano_2002`=NULL WHERE `ano_2002`=0.0;
14 UPDATE `idh` SET `ano_2003`=NULL WHERE `ano_2003`=0.0;
15 UPDATE `idh` SET `ano_2004`=NULL WHERE `ano_2004`=0.0;
16 UPDATE `idh` SET `ano_2005`=NULL WHERE `ano_2005`=0.0;
17 UPDATE `idh` SET `ano_2006`=NULL WHERE `ano_2006`=0.0;
18 UPDATE `idh` SET `ano_2007`=NULL WHERE `ano_2007`=0.0;
19 UPDATE `idh` SET `ano_2008`=NULL WHERE `ano_2008`=0.0;
20 UPDATE `idh` SET `ano_2009`=NULL WHERE `ano_2009`=0.0;
21 UPDATE `idh` SET `ano_2010`=NULL WHERE `ano_2010`=0.0;
22 UPDATE `idh` SET `ano_2011`=NULL WHERE `ano_2011`=0.0;
23 UPDATE `idh` SET `ano_2012`=NULL WHERE `ano_2012`=0.0;
24 UPDATE `idh` SET `ano_2013`=NULL WHERE `ano_2013`=0.0;
25 UPDATE `idh` SET `ano_2014`=NULL WHERE `ano_2014`=0.0;
26 UPDATE `idh` SET `ano_2015`=NULL WHERE `ano_2015`=0.0;
27 UPDATE `idh` SET `ano_2016`=NULL WHERE `ano_2016`=0.0;
28 UPDATE `idh` SET `ano_2017`=NULL WHERE `ano_2017`=0.0;
29 UPDATE `idh` SET `ano_2018`=NULL WHERE `ano_2018`=0.0;
30 UPDATE `idh` SET `ano_2019`=NULL WHERE `ano_2019`=0.0;
```

Ao fim da verificação dos registros, as colunas com registros datados anteriormente e posteriormente ao período da pesquisa foram removidas da base de dados.

```

1   ALTER TABLE idh
2   DROP COLUMN ano_1990,
3   DROP COLUMN ano_1991,
4   DROP COLUMN ano_1992,
5   DROP COLUMN ano_1993,
6   DROP COLUMN ano_1994,
7   DROP COLUMN ano_1995,
8   DROP COLUMN ano_1996,
9   DROP COLUMN ano_1997,
10  DROP COLUMN ano_1998,
11  DROP COLUMN ano_1999,
12  DROP COLUMN ano_2000,
13  DROP COLUMN ano_2001,
14  DROP COLUMN ano_2002,
15  DROP COLUMN ano_2003,
16  DROP COLUMN ano_2004,
17  DROP COLUMN ano_2005,
18  DROP COLUMN ano_2006,
19  DROP COLUMN ano_2007,
20  DROP COLUMN ano_2008,
21  DROP COLUMN ano_2009,
22  DROP COLUMN ano_2019;|

```

Ao final das alterações é possível visualizar as informações da tabela.

| Field | Type | Null | Key | Default | Extra |
|-----------|---------------|------|-----|---------|----------------|
| idh_rank | int(3) | YES | | NULL | |
| id | int(3) | NO | PRI | NULL | auto_increment |
| nome_pais | varchar(96) | YES | | NULL | |
| iso | varchar(2) | YES | | NULL | |
| ano_2010 | decimal(10,3) | YES | | NULL | |
| ano_2011 | decimal(10,3) | YES | | NULL | |
| ano_2012 | decimal(10,3) | YES | | NULL | |
| ano_2013 | decimal(10,3) | YES | | NULL | |
| ano_2014 | decimal(10,3) | YES | | NULL | |
| ano_2015 | decimal(10,3) | YES | | NULL | |
| ano_2016 | decimal(10,3) | YES | | NULL | |
| ano_2017 | decimal(10,3) | YES | | NULL | |
| ano_2018 | decimal(10,3) | YES | | NULL | |

5.3. Tabela desemprego

A relação de dados *desemprego* foi processada de forma semelhante a tabela *idh*.

Primeiramente, os registros foram inseridos na relação diretamente de um arquivo CSV gerado anteriormente.

```
✓ 191 linha(s) inseridas. (A consulta demorou 0,0094 segundos)

LOAD DATA INFILE "C:/Users/Spaox/Documents/Projeto/Arquivos/Tabelas/Tratadas/desemprego.csv" INTO TABLE desemprego character set latin1 FIELDS
TERMINATED BY ",";
[ Editar em linha ] [ Edita ] [ Criar código PHP ]
```

Em seguida, os campos *id* e *iso* também foram adicionados à tabela.

Uma função *join* utilizando o nome dos territórios como referência serviu para atualizar os registros do campo *iso*. Devido a similaridade dos nomes dos países encontrados entre as duas tabelas, foram utilizadas as informações da tabela *idh*.

Ao final, as colunas que não seriam úteis para a pesquisa foram removidas.

Através do comando *describe* é possível ver as informações da tabela, como nome dos campos, tipos e relações.

Ao final das alterações é possível visualizar as informações da tabela.

| Field | Type | Null | Key | Default | Extra |
|-----------|---------------|------|-----|---------|----------------|
| idh_rank | int(3) | YES | | NULL | |
| id | int(3) | NO | PRI | NULL | auto_increment |
| nome_pais | varchar(96) | YES | | NULL | |
| iso | varchar(2) | YES | | NULL | |
| ano_2010 | decimal(10,1) | YES | | NULL | |
| ano_2011 | decimal(10,1) | YES | | NULL | |
| ano_2012 | decimal(10,1) | YES | | NULL | |
| ano_2013 | decimal(10,1) | YES | | NULL | |
| ano_2014 | decimal(10,1) | YES | | NULL | |
| ano_2015 | decimal(10,1) | YES | | NULL | |
| ano_2016 | decimal(10,1) | YES | | NULL | |
| ano_2017 | decimal(10,1) | YES | | NULL | |
| ano_2018 | decimal(10,1) | YES | | NULL | |

5.4. Tabela educacao

A relação de dados *educacao* foi processada de maneira semelhante a tabela *idh* e a tabela *desemprego*.

Inicialmente, os registros foram inseridos na relação de dados diretamente de um arquivo CSV.

```

    ✓ 189 linha(s) inseridas. (A consulta demorou 0,0094 segundos.)

LOAD DATA INFILE "C:/Users/Spoox/Documents/Projeto/Arquivos/Tabelas/Tratadas/educacao.csv" INTO TABLE educacao character set latin1 FIELDS
TERMINATED BY ",";
[ Editar em linha ] [ Edita ] [ Criar código PHP ]

```

Em seguida, os campos *id* e *iso* também foram adicionados à tabela.

Uma função *join* utilizando o nome dos territórios como referência serviu para atualizar os registros do campo *iso*.

Depois da atualização dos campos *iso*, registros com valores “0.0” foram atualizados para “NULL”, já que registros nulos foram inseridos com valores “0.0” na tabela.

Ao final, as colunas que não seriam úteis para a pesquisa foram removidas.

Ao final das alterações é possível visualizar as informações da tabela.

| Field | Type | Null | Key | Default | Extra |
|------------|---------------|------|-----|---------|----------------|
| idh_rank | int(3) | YES | | NULL | |
| <i>id</i> | int(3) | NO | PRI | NULL | auto_increment |
| nome_pais | varchar(96) | YES | | NULL | |
| <i>iso</i> | varchar(2) | YES | | NULL | |
| ano_2010 | decimal(10,3) | YES | | NULL | |
| ano_2011 | decimal(10,3) | YES | | NULL | |
| ano_2012 | decimal(10,3) | YES | | NULL | |
| ano_2013 | decimal(10,3) | YES | | NULL | |
| ano_2014 | decimal(10,3) | YES | | NULL | |
| ano_2015 | decimal(10,3) | YES | | NULL | |
| ano_2016 | decimal(10,3) | YES | | NULL | |
| ano_2017 | decimal(10,3) | YES | | NULL | |
| ano_2018 | decimal(10,3) | YES | | NULL | |

5.5. Tabela *expectativa_vida*

A relação de dados *expectativa_vida* foi processada da mesma forma que as tabelas anteriores.

Primeiramente, os registros foram inseridos na relação de dados diretamente de um arquivo CSV.

```

    ✓ 191 linha(s) inseridas. (A consulta demorou 0,0110 segundos.)

LOAD DATA INFILE "C:/Users/Spoox/Documents/Projeto/Arquivos/Tabelas/Tratadas/expectativa_vida.csv" INTO TABLE expectativa_vida character set latin1
FIELDS TERMINATED BY ",";
[ Editar em linha ] [ Edita ] [ Criar código PHP ]

```

Em seguida, as colunas *id* e *iso* foram adicionadas à tabela.

Uma função *join* utilizando o nome dos territórios como referência serviu para atualizar os registros do campo *iso*. A tabela *desemprego* foi utilizada como auxiliar para a realização da função *join* já que ambas possuíam o mesmo número de registros.

Após as atualizações, os registros com valores “0.0” foram atualizados para “NULL”, já que registros nulos foram inseridos com valores “0.0” na tabela.

As colunas que não seriam úteis para a pesquisa foram removidas.

Ao final das alterações é possível visualizar as informações da tabela.

| Field | Type | Null | Key | Default | Extra |
|-----------|---------------|------|-----|---------|----------------|
| idh_rank | int(3) | YES | | NULL | |
| id | int(3) | NO | PRI | NULL | auto_increment |
| nome_pais | varchar(96) | YES | | NULL | |
| iso | varchar(2) | YES | | NULL | |
| ano_2010 | decimal(10,1) | YES | | NULL | |
| ano_2011 | decimal(10,1) | YES | | NULL | |
| ano_2012 | decimal(10,1) | YES | | NULL | |
| ano_2013 | decimal(10,1) | YES | | NULL | |
| ano_2014 | decimal(10,1) | YES | | NULL | |
| ano_2015 | decimal(10,1) | YES | | NULL | |
| ano_2016 | decimal(10,1) | YES | | NULL | |
| ano_2017 | decimal(10,1) | YES | | NULL | |
| ano_2018 | decimal(10,1) | YES | | NULL | |

Ao fim do processamento e inserção dos dados das informações do Relatório de Desenvolvimento Humano da ONU foram inseridos e tratados os dados referentes aos Indicadores de Governança Mundial do Banco Mundial.

De maneira semelhante as tabelas do Relatório de Desenvolvimento Humano da ONU, os processos serão descritos, porém apenas a primeira tabela referenciada conterá imagens dos processos realizados.

5.6. Tabela *estabilidade_politica*

A primeira tabela da categoria a ser processada e tratada foi a tabela *estabilidade_politica*. Após a criação da tabela, os registros dispostos no arquivo CSV foram inseridos diretamente na relação de dados.

```
✓ 214 linha(s) inseridas. (A consulta demorou 0,0102 segundos.)

LOAD DATA INFILE "C:/Users/Spaox/Documents/Projeto/Arquivos/Tabelas/Tratadas/estabilidade_politica.csv" INTO TABLE estabilidade_politica character
set latin1 FIELDS TERMINATED BY ";"';

[ Editar em linha ] [ Edita ] [ Criar código PHP ]
```

Em seguida, as colunas *id* e *iso* foram adicionadas à tabela.

```
✓ MySQL não retornou nenhum registo. (A consulta demorou 0,0416 segundos.)

ALTER TABLE estabilidade_politica ADD COLUMN( id INT(3) AUTO_INCREMENT PRIMARY KEY, iso VARCHAR(2) );

[ Editar em linha ] [ Edita ] [ Criar código PHP ]
```

Para atualizar os registros da nova coluna, duas funções *join* foram realizadas. Utilizando o nome dos territórios como referência, as tabelas *expectativa_vida* e *paises* foram usadas como auxiliares durante o desenvolvimento da função.

```
✓ 28 linha(s) afectadas. (A consulta demorou 0,0576 segundos.)

UPDATE estabilidade_politica INNER JOIN paises ON estabilidade_politica.nome_pais = paises.nome_pais SET estabilidade_politica.iso = paises.iso;

[ Editar em linha ] [ Edita ] [ Criar código PHP ]
```

```
✓ 165 linha(s) afectadas. (A consulta demorou 0,0679 segundos.)

UPDATE estabilidade_politica INNER JOIN expectativa_vida ON estabilidade_politica.nome_pais = expectativa_vida.nome_pais SET estabilidade_politica.iso =
= expectativa_vida.iso;

[ Editar em linha ] [ Edita ] [ Criar código PHP ]
```

Após uma verificação, alguns dados ainda permaneceram nulos, foi necessário então, preenchê-los manualmente.

Depois da atualização dos registros do campo *iso*, os registros percentuais com valores “-1” foram atualizados para “NULL”, dessa forma seriam desconsiderados em futuros cálculos. Os registros que levavam os valores “-1” haviam sido alterados ainda no Microsoft Excel.

```
1 UPDATE estabilidade_politica SET perc_ano_2010 = NULL WHERE perc_ano_2010 = -1;
2 UPDATE estabilidade_politica SET perc_ano_2011 = NULL WHERE perc_ano_2011 = -1;
3 UPDATE estabilidade_politica SET perc_ano_2012 = NULL WHERE perc_ano_2012 = -1;
4 UPDATE estabilidade_politica SET perc_ano_2013 = NULL WHERE perc_ano_2013 = -1;
5 UPDATE estabilidade_politica SET perc_ano_2014 = NULL WHERE perc_ano_2014 = -1;
6 UPDATE estabilidade_politica SET perc_ano_2015 = NULL WHERE perc_ano_2015 = -1;
7 UPDATE estabilidade_politica SET perc_ano_2016 = NULL WHERE perc_ano_2016 = -1;
8 UPDATE estabilidade_politica SET perc_ano_2017 = NULL WHERE perc_ano_2017 = -1;
9 UPDATE estabilidade_politica SET perc_ano_2018 = NULL WHERE perc_ano_2018 = -1;
```

Ao final das alterações é possível visualizar as informações da tabela.

| Field | Type | Null | Key | Default | Extra |
|---------------|---------------|------|-----|---------|----------------|
| id | int(3) | NO | PRI | NULL | auto_increment |
| nome_pais | varchar(96) | YES | | NULL | |
| iso | varchar(2) | YES | | NULL | |
| perc_ano_2010 | decimal(10,2) | YES | | NULL | |
| perc_ano_2011 | decimal(10,2) | YES | | NULL | |
| perc_ano_2012 | decimal(10,2) | YES | | NULL | |
| perc_ano_2013 | decimal(10,2) | YES | | NULL | |
| perc_ano_2014 | decimal(10,2) | YES | | NULL | |
| perc_ano_2015 | decimal(10,2) | YES | | NULL | |
| perc_ano_2016 | decimal(10,2) | YES | | NULL | |
| perc_ano_2017 | decimal(10,2) | YES | | NULL | |
| perc_ano_2018 | decimal(10,2) | YES | | NULL | |

5.7. Tabela controle_corrupcao

De forma semelhante a tabela anterior, a relação de dados *controle_corrupcao* foi tratada e processada após receber registros de um arquivo CSV.

```
✓ 214 linha(s) inseridas. (A consulta demorou 0.0110 segundos.)  

LOAD DATA INFILE "C:/Users/Spaox/Documents/Projeto/Arquivos/Tabelas/Tratadas/controle_corrupcao.csv" INTO TABLE controle_corrupcao character set  

latin1 FIELDS TERMINATED BY ";"  

[ Editar em linha ] [ Edita ] [ Criar código PHP ]
```

Após a inserção dos arquivos, duas novas colunas foram adicionadas à tabela: *id* e *iso*.

Utilizando a relação de dados *estabilidade_politica* como auxiliar, uma função *join* referenciando o nome dos territórios semelhantes entre as tabelas atualizou os registros do campo *iso*.

Depois da atualização dos campos *iso*, os campos com valores “-1” foram atualizados para “NULL”, dessa forma seriam desconsiderados para cálculos futuros. Os registros que levavam os valores “-1” haviam sido alterados ainda no Microsoft Excel.

Ao final das alterações é possível visualizar as informações da tabela.

| Field | Type | Null | Key | Default | Extra |
|---------------|---------------|------|-----|---------|----------------|
| id | int(3) | NO | PRI | NULL | auto_increment |
| nome_pais | varchar(96) | YES | | NULL | |
| iso | varchar(2) | YES | | NULL | |
| perc_ano_2010 | decimal(10,2) | YES | | NULL | |
| perc_ano_2011 | decimal(10,2) | YES | | NULL | |
| perc_ano_2012 | decimal(10,2) | YES | | NULL | |
| perc_ano_2013 | decimal(10,2) | YES | | NULL | |
| perc_ano_2014 | decimal(10,2) | YES | | NULL | |
| perc_ano_2015 | decimal(10,2) | YES | | NULL | |
| perc_ano_2016 | decimal(10,2) | YES | | NULL | |
| perc_ano_2017 | decimal(10,2) | YES | | NULL | |
| perc_ano_2018 | decimal(10,2) | YES | | NULL | |

5.8. Tabela efetividade_governo

Seguindo o padrão para a categoria dos Indicadores de Governança Mundial do Banco Mundial, a tabela *efetividade_governo* recebeu os dados de um arquivo CSV após ser criada.

```
✓ 214 linha(s) inseridas. (A consulta demorou 0,0127 segundos.)  
LOAD DATA INFILE "C:/Users/Spaox/Documents/Projeto/Arquivos/Tabelas/Tratadas/efetividade_governo.csv" INTO TABLE efetividade_governo character set latin1 FIELDS TERMINATED BY ";"  
[ Editar em linha ] [ Edita ] [ Criar código PHP ]
```

O padrão seguiu para a atualização da tabela. Assim, as colunas *id* e *iso* também foram adicionadas.

Como referência para atualização dos registros do campo *iso*, a relação de dados *estabilidade_politica* foi utilizada para realização da função *join*.

Depois da atualização dos campos *iso*, os campos com valores “-1” foram atualizados para “NULL”. Os registros que levavam os valores “-1” haviam sido alterados ainda no Microsoft Excel.

Ao final das alterações é possível visualizar as informações da tabela.

| Field | Type | Null | Key | Default | Extra |
|---------------|---------------|------|-----|---------|----------------|
| id | int(3) | NO | PRI | NULL | auto_increment |
| nome_pais | varchar(96) | YES | | NULL | |
| iso | varchar(2) | YES | | NULL | |
| perc_ano_2010 | decimal(10,2) | YES | | NULL | |
| perc_ano_2011 | decimal(10,2) | YES | | NULL | |
| perc_ano_2012 | decimal(10,2) | YES | | NULL | |
| perc_ano_2013 | decimal(10,2) | YES | | NULL | |
| perc_ano_2014 | decimal(10,2) | YES | | NULL | |
| perc_ano_2015 | decimal(10,2) | YES | | NULL | |
| perc_ano_2016 | decimal(10,2) | YES | | NULL | |
| perc_ano_2017 | decimal(10,2) | YES | | NULL | |
| perc_ano_2018 | decimal(10,2) | YES | | NULL | |

5.9. Tabela divisao_etnica

A relação de dados contendo registros de índice de fracionalização étnica dos territórios foi tratada e processada de maneira simples. Após a criação da relação de dados, os registros do arquivo CSV foram inseridos na tabela *divisao_etnica*.

```
✓ 8808 linha(s) inseridas. (A consulta demorou 0,0418 segundos.)

LOAD DATA INFILE "C:/Users/Spaox/Documents/Projeto/Arquivos/Tabelas/Tratadas/divisao_etnica.csv" INTO TABLE divisao_etnica character set latin1
FIELDS TERMINATED BY ",";

[ Editar em linha ] [ Edita ] [ Criar código PHP ]
```

Assim como nas relações anteriores, as colunas *id* e *iso* foram adicionadas.

```
✓ MySQL não retornou nenhum registo. (A consulta demorou 0,0373 segundos.)

ALTER TABLE divisao_etnica ADD COLUMN( id INT(3) AUTO_INCREMENT PRIMARY KEY, iso VARCHAR(2) );

[ Editar em linha ] [ Edita ] [ Criar código PHP ]
```

Utilizando as tabelas *paises* e *estabilidade_politica* como auxiliares os registros do campo *iso* foram atualizados através de uma função *join*.

```
✓ 12 linha(s) afectadas. (A consulta demorou 0,0326 segundos.)

UPDATE divisao_etnica INNER JOIN paises ON divisao_etnica.nome_pais = paises.nome_pais SET divisao_etnica.iso = paises.iso;

[ Editar em linha ] [ Edita ] [ Criar código PHP ]

✓ 135 linha(s) afectadas. (A consulta demorou 0,0323 segundos.)

UPDATE divisao_etnica INNER JOIN estabilidade_politica ON divisao_etnica.nome_pais = estabilidade_politica.nome_pais SET divisao_etnica.iso = estabilidade_politica.iso;

[ Editar em linha ] [ Edita ] [ Criar código PHP ]
```

Os registros que não foram atualizados após as funções *join* foram atualizados manualmente.

Após a atualização dos registros no campo *iso*, os dados com datas não correspondentes ao período de estudo foram removidos da relação.

```
✓ 8652 linha(s) afectadas. (A consulta demorou 0,0485 segundos.)

DELETE FROM `divisao_etnica` WHERE ano != 2010;

[ Editar em linha ] [ Edita ] [ Criar código PHP ]
```

Os registros de índice de fracionalização étnica foram copiados e reproduzidos para todos os anos do período pesquisado.

Ao final das alterações é possível visualizar as informações da tabela.

| Field | Type | Null | Key | Default | Extra |
|-----------|---------------|------|-----|---------|----------------|
| id | int(3) | NO | PRI | NULL | auto_increment |
| nome_pais | varchar(96) | YES | | NULL | |
| iso | varchar(2) | YES | | NULL | |
| ano | int(4) | YES | | NULL | |
| indice | decimal(10,3) | YES | | NULL | |

5.10. Tabela regime_politico

Antes da criação da tabela alguns atributos foram removidos no Microsoft Excel devido a sua falta de utilidade para a pesquisa. Em seguida, os registros foram inseridos diretamente na relação de dados.

```
✓ 52896 linha(s) inseridas. (A consulta demorou 0,4833 segundos.)

LOAD DATA INFILE "C:/Users/Spaox/Documents/Projeto/Arquivos/Tabelas/Tratadas/regime_politico.csv" INTO TABLE regime_politico character set latin1 FIELDS TERMINATED BY
";";
[ Editar em linha ] [ Edita ] [ Criar código PHP ]
```

Após a inserção dos dados, duas novas colunas foram adicionadas à tabela *regime_politico*: as colunas *id* e *iso*.

```
✓ MySQL não retornou nenhum registo. (A consulta demorou 0,0644 segundos.)

ALTER TABLE regime_politico ADD COLUMN(id INT(6) AUTO_INCREMENT PRIMARY KEY, ISO VARCHAR(2) NULL );
[ Editar em linha ] [ Edita ] [ Criar código PHP ]
```

Logo após a adição das duas novas colunas, uma função *join* foi utilizada para atualizar os campos *iso* com dados da tabela *países* utilizando o nome dos territórios como referência.

```
✓ 1674 linha(s) afectadas. (A consulta demorou 0,5721 segundos.)

UPDATE regime_politico INNER JOIN paises ON regime_politico.nome_pais = paises.nome_pais SET regime_politico.iso = paises.iso;
[ Editar em linha ] [ Edita ] [ Criar código PHP ]
```

Os registros que não foram atualizados após as funções *join* foram atualizados manualmente.

Com todos os atributos preenchidos, os registros foram checados para encontrar dados repetidos e problemas de espaçamento nos campos inseridos. Alguns campos do atributo *descricao_regime* possuíam quebras de linha que dificultavam a leitura das informações, para isso, foi necessário remover essas quebras de linha.

```
✓ 52896 linha(s) afectadas. (A consulta demorou 1,0494 segundos.)  
UPDATE regime_politico SET descricao_regime = TRIM(REPLACE(REPLACE(REPLACE(descricao_regime,'\'t',''),'\n',''),'\r',''));  
[ Editar em linha ] [ Edita ] [ Criar código PHP ]
```

Após a verificação dos registros faltantes, novas alterações foram necessárias. Primeiramente, identificou-se a necessidade de alterar alguns registros do atributo *codigo_regime* onde os campos *descricao_regime* eram iguais a “no data”. Essa alteração foi feita porque registros nulos foram inseridos com o valor “0” nos campos *codigo_regime*, porém já haviam registros com o valor “0” que indicavam o sistema político “closed autocracy” nos campos *descricao_regime*.

```
✓ 22687 linha(s) afectadas. (A consulta demorou 0,1842 segundos.)  
UPDATE regime_politico SET codigo_regime = 99 WHERE descricao_regime = "no data";  
[ Editar em linha ] [ Edita ] [ Criar código PHP ]
```

Logo após a verificação dos dados, utilizou-se funções de agregação para identificar valores no campo “ano” maiores e menores que os do período estudado. Após a identificação, os registros foram excluídos.

```
✓ 50388 linha(s) afectadas. (A consulta demorou 0,1724 segundos.)  
DELETE FROM regime_politico WHERE ano BETWEEN 1789 AND 2009;  
[ Editar em linha ] [ Edita ] [ Criar código PHP ]
```

```
✓ 456 linha(s) afectadas. (A consulta demorou 0,0117 segundos.)  
DELETE FROM regime_politico WHERE ano > 2018;  
[ Editar em linha ] [ Edita ] [ Criar código PHP ]
```

Em seguida, houve uma verificação dos códigos dos regimes políticos e suas respectivas descrições. Novamente foi necessário atualizar alguns registros da tabela, dessa vez registros do campo “*descricao_regime*” foram atualizados de acordo com os campos “*codigo_regime*”. A verificação foi realizada para as quatro categorias de sistemas políticos destacados.

```
✓ 514 linha(s) afectadas. (A consulta demorou 0,0130 segundos.)  
UPDATE regime_politico SET descricao_regime = "electoral democracy" WHERE codigo_regime = 2;  
[ Editar em linha ] [ Edita ] [ Criar código PHP ]
```

```
✓ 348 linha(s) afectadas. (A consulta demorou 0,0119 segundos.)  
UPDATE regime_politico SET descricao_regime = "liberal democracy" WHERE codigo_regime = 3;  
[ Editar em linha ] [ Edita ] [ Criar código PHP ]
```

Ao final das alterações é possível visualizar as informações da tabela.

| Field | Type | Null | Key | Default | Extra |
|------------------|-------------|------|-----|---------|----------------|
| id | int(6) | NO | PRI | NULL | auto_increment |
| nome_pais | varchar(96) | YES | | NULL | |
| ISO | varchar(2) | YES | | NULL | |
| ano | int(4) | YES | | NULL | |
| codigo_regime | int(2) | YES | | NULL | |
| descricao_regime | varchar(57) | YES | | NULL | |

5.11. Tabela terrorismo

Grande parte do tratamento para a tabela terrorismo foi realizado no Microsoft Excel, em especial, a identificação dos atributos previsores e do atributo classe. Assim, os registros foram inseridos na relação de dados e em seguida tratados.

```
✓ 201183 linha(s) inseridas. (A consulta demorou 4,0972 segundos.)  
LOAD DATA INFILE "C:/Users/Spaox/Documents/Projeto/Arquivos/Tabelas/Tratadas/terrorismo.csv" INTO TABLE terrorismo character set latin1 FIELDS TERMINATED BY ";"  
[ Editar em linha ] [ Edita ] [ Criar código PHP ]
```

Após a inserção dos dados, duas novas colunas foram adicionadas à tabela: as colunas *id* e *iso*.

```
✓ MySQL não retornou nenhum registo. (A consulta demorou 1,6672 segundos.)  
ALTER TABLE terrorismo ADD COLUMN( id INT(5) AUTO_INCREMENT PRIMARY KEY, iso VARCHAR(2) );  
[ Editar em linha ] [ Edita ] [ Criar código PHP ]
```

Logo após a adição das duas novas colunas, uma função *join* foi utilizada para atualizar os campos *iso* com dados da tabela *paises* utilizando o nome dos países como referência.

```
✓ 90755 linha(s) afectadas. (A consulta demorou 100,2388 segundos.)  
UPDATE terrorismo INNER JOIN paises ON terrorismo.nome_pais = paises.nome_pais SET terrorismo.iso = paises.iso;  
[ Editar em linha ] [ Edita ] [ Criar código PHP ]
```

Os registros que não foram atualizados após as funções *join* foram atualizados manualmente.

Ao fim da criação das colunas para registros identificadores, duas novas colunas foram criadas: *houve_obitos* e *houve_feridos*. Essas colunas receberam valores binários, indicando 0 para registros onde não houveram óbitos ou feridos, e 1 para registros onde houveram óbitos ou feridos.

```
✓ MySQL não retornou nenhum registo. (A consulta demorou 0,0229 segundos.)  
ALTER TABLE terrorismo ADD COLUMN( houve_obitos INT(1), houve_feridos INT(1) );  
[ Editar em linha ] [ Edita ] [ Criar código PHP ]
```

Uma nova coluna foi criada para identificação dos grupos perpretadores. Utilizando a coluna *grupo_terrorista* como referência foram atualizados os valores da

nova coluna criada: *identificacao_grupo*. Os registros onde houve reconhecimento do grupo perpretador receberam o valor 1 no campo *identificacao_grupo*, já os registros que marcaram o valor “Unknown” receberam o valor 0.

```
✓ MySQL não retornou nenhum registro. (A consulta demorou 0.0290 segundos.)  
alter table terrorismo add COLUMN( identificacao_grupo INT(1) );  
[ Editar em linha ] [ Edita ] [ Criar código PHP ]
```

Após a criação dos novos campos e a atualização dos registros, os dados que não seriam utilizados para a pesquisa foram removidos. Primeiramente, dados das regiões dos continentes americano, europeu e da Oceania foram excluídos.

```
✓ 7049 linha(s) afectadas. (A consulta demorou 0.5373 segundos.)  
DELETE FROM `terrorismo` WHERE codigo_regiao = 1 OR codigo_regiao = 2 OR codigo_regiao = 3 OR codigo_regiao = 8 OR codigo_regiao = 9 OR codigo_regiao = 12;  
[ Editar em linha ] [ Edita ] [ Criar código PHP ]
```

Em seguida, os dados que não faziam parte do período estudado foram removidos.

```
✓ 94884 linha(s) afectadas. (A consulta demorou 1,2828 segundos.)  
DELETE FROM `terrorismo` WHERE ano BETWEEN 1970 AND 2009;  
[ Editar em linha ] [ Edita ] [ Criar código PHP ]
```

```
✓ 8495 linha(s) afectadas. (A consulta demorou 0,3914 segundos.)  
DELETE FROM `terrorismo` WHERE ano > 2018;  
[ Editar em linha ] [ Edita ] [ Criar código PHP ]
```

Houve, em seguida, uma verificação para encontrar linhas idênticas na relação dos dados e compará-las

Ao final das alterações é possível visualizar as informações da tabela.

| Field | Type | Null | Key | Default | Extra |
|-----------------------|--------------|------|-----|---------|----------------|
| id | int(5) | NO | PRI | NULL | auto_increment |
| id_evento | varchar(12) | YES | | NULL | |
| ano | int(4) | YES | | NULL | |
| mes | int(2) | YES | | NULL | |
| dia | int(2) | YES | | NULL | |
| nome_continente | varchar(10) | YES | | NULL | |
| codigo_pais | int(4) | YES | | NULL | |
| nome_pais | varchar(96) | YES | | NULL | |
| iso | varchar(2) | YES | | NULL | |
| codigo_regiao | int(2) | YES | | NULL | |
| nome_regiao | varchar(30) | YES | | NULL | |
| provincia | varchar(96) | YES | | NULL | |
| cidade | varchar(96) | YES | | NULL | |
| criterio1 | int(1) | YES | | NULL | |
| criterio2 | int(1) | YES | | NULL | |
| criterio3 | int(1) | YES | | NULL | |
| duvida_terrorismo | int(1) | YES | | NULL | |
| incidente_alternativo | varchar(30) | YES | | NULL | |
| ataques_conectados | int(1) | YES | | NULL | |
| sucesso_ataque | int(1) | YES | | NULL | |
| codigo_metodo_ataque | int(1) | YES | | NULL | |
| metodo_ataque | varchar(35) | YES | | NULL | |
| codigo_tipo_vitima | int(2) | YES | | NULL | |
| tipo_vitima | varchar(35) | YES | | NULL | |
| codigo_subtipo_vitima | int(3) | YES | | NULL | |
| subtipo_vitima | varchar(100) | YES | | NULL | |
| codigo_nac_alvo | int(4) | YES | | NULL | |
| nac_alvo | varchar(96) | YES | | NULL | |
| grupo_terrorista | varchar(100) | YES | | NULL | |
| identificacao_grupo | int(1) | YES | | NULL | |
| reivindicacao_ataque | int(1) | YES | | NULL | |
| codigo_tipo_arma | int(2) | YES | | NULL | |
| tipo_arma | varchar(30) | YES | | NULL | |
| numero_obitos | int(4) | YES | | NULL | |
| houve_obitos | int(1) | YES | | NULL | |
| numero_feridos | int(4) | YES | | NULL | |
| houve_feridos | int(1) | YES | | NULL | |
| danos_propriedades | int(1) | YES | | NULL | |

Todas as tabelas foram previamente tratadas para que a tabela final contendo as informações de todos os territórios pesquisados fosse criada e alimentada.

5.12. Tabela perfil_paises

Assim que todas as relações passaram pelas tarefas de processamento de dados, a tabela *perfil_paises* foi criada. O objetivo para criação dessa tabela foi reunir todas as informações anteriormente recolhidas e criar uma visualização para essas informações, além do trabalho em conjunto com a tabela *terrorismo* para elaboração dos modelos de *machine learning*.

Primeiramente a tabela foi criada.

```
MySQL não retornou nenhum registo. (A consulta demorou 0,0782 segundos.)  
  
CREATE TABLE `perfil_paises` ( `nome_continente` varchar(10) COLLATE utf8_swedish_ci DEFAULT NULL, `codigo_regiao_gtd` int(2) DEFAULT NULL, `nome_regiao` varchar(30) COLLATE utf8_swedish_ci DEFAULT NULL, `representacao_pais` varchar(20) COLLATE utf8_swedish_ci DEFAULT NULL, `nome_pais` varchar(96) COLLATE utf8_swedish_ci DEFAULT NULL, `iso` varchar(2) COLLATE utf8_swedish_ci DEFAULT NULL, `ano` int(4) DEFAULT NULL, `regime_politico` varchar(57) COLLATE utf8_swedish_ci DEFAULT NULL, `divisao_etnica` decimal(10,3) DEFAULT NULL, `indice_divisao_etnica` varchar(20) COLLATE utf8_swedish_ci DEFAULT NULL, `religiao_predominante` varchar(20) COLLATE utf8_swedish_ci DEFAULT NULL, `religiao_secundaria` varchar(20) COLLATE utf8_swedish_ci DEFAULT NULL, `indice_desemprego` decimal(10,1) DEFAULT NULL, `indice_desemprego_regiao` varchar(20) COLLATE utf8_swedish_ci [ Edita ]
```

Em seguida, a tabela começou a receber informações das outras relações utilizando diversas funções *join*. O campo *iso* foi utilizado como referência para a combinação dos registros.

Primeiramente, os dados da tabela *paises* foram inseridos na tabela *perfil_paises*. Utilizando o campo *iso* como referência, as informações de continente, código da região, nome do país e código ISO foram inseridas na tabela.

```
258 linha(s) inseridas. (A consulta demorou 0,0154 segundos.)  
  
INSERT INTO perfil_paises (nome_continente, nome_regiao, codigo_regiao_gtd, nome_pais, iso) SELECT paises.nome_continente, paises.nome_regiao, paises.codigo_regiao_gtd, paises.nome_pais, paises.iso FROM paises;  
[ Editar em linha ] [ Edita ] [ Criar código PHP ]
```

Em seguida, a tabela recebeu informações das relações referentes aos índices do Relatório de Desenvolvimento Humano da ONU.

Inicialmente o campo *indice_idh* foi atualizado com os registros da tabela *idh*.

```
188 linha(s) afectadas. (A consulta demorou 0,1121 segundos.)  
  
UPDATE perfil_paises INNER JOIN idh ON perfil_paises.iso = idh.iso AND (perfil_paises.ano = 2010 AND idh.ano_2010) SET perfil_paises.indice_idh = idh.ano_2010;  
[ Editar em linha ] [ Edita ] [ Criar código PHP ]
```

Depois, o campo *indice_desemprego* foi atualizado com os registros da tabela *desemprego*.

```
180 linha(s) afectadas. (A consulta demorou 0,9474 segundos.)  
  
UPDATE perfil_paises INNER JOIN desemprego ON perfil_paises.iso = desemprego.iso AND (perfil_paises.ano = 2010 AND desemprego.ano_2010) SET perfil_paises.indice_desemprego = desemprego.ano_2010;  
[ Editar em linha ] [ Edita ] [ Criar código PHP ]
```

Em seguida, o campo *indice_educacao* foi atualizado com os registros da tabela *educacao*.

```
✓ 188 linha(s) afectadas. (A consulta demorou 0,1816 segundos.)  
UPDATE perfil_paises INNER JOIN educacao ON perfil_paises.iso = educacao.iso AND (perfil_paises.ano = 2010 AND educacao.ano_2010) SET perfil_paises.indice_educacao = educacao.ano_2010;  
[ Editar em linha ] [ Edita ] [ Criar código PHP ]
```

E por fim o campo *indice_expvida* foi atualizado com os registros da tabela *expectativa_vida*.

```
✓ 191 linha(s) afectadas. (A consulta demorou 1,7608 segundos.)  
UPDATE perfil_paises INNER JOIN expectativa_vida ON perfil_paises.iso = expectativa_vida.iso AND (perfil_paises.ano = 2010 AND expectativa_vida.ano_2010) SET perfil_paises.indice_expvida = expectativa_vida.ano_2010;  
[ Editar em linha ] [ Edita ] [ Criar código PHP ]
```

De maneira semelhante à atualização dos campos referentes ao Relatório de Desenvolvimento Humano, os campos Indicadores de Governança Mundial do Banco Mundial foram atualizados na tabela *perfil_paises*.

Inicialmente, o campo *indice_ctrlcorrupcao* foi atualizado com os registros da tabela *controle_corrupcao*.

```
✓ 210 linha(s) afectadas. (A consulta demorou 1,5114 segundos.)  
UPDATE perfil_paises INNER JOIN controle_corrupcao ON perfil_paises.iso = controle_corrupcao.iso AND (perfil_paises.ano = 2010 AND controle_corrupcao.perc_ano_2010) SET perfil_paises.indice_ctrlcorrupcao = controle_corrupcao.perc_ano_2010;  
[ Editar em linha ] [ Edita ] [ Criar código PHP ]
```

Após receber os valores, o campo *indice_ctrlcorrupcao* passou a se chamar *indice_ctrlcorrupcao_rank*.

Depois, o campo *indice_efetividadegov* foi atualizado com os registros da tabela *efetividade_governo*.

```
✓ 209 linha(s) afectadas. (A consulta demorou 1,1207 segundos.)  
UPDATE perfil_paises INNER JOIN efetividade_governo ON perfil_paises.iso = efetividade_governo.iso AND (perfil_paises.ano = 2010 AND efetividade_governo.perc_ano_2010) SET perfil_paises.indice_efetividadegov = efetividade_governo.perc_ano_2010;  
[ Editar em linha ] [ Edita ] [ Criar código PHP ]
```

Após receber os valores, o campo *indice_efetividadegov* passou a se chamar *indice_efetividadegov_rank*.

E por fim, o campo *indice_estpolitica* foi atualizado com os registros da tabela *estabilidade_politica*.

```
✓ 211 linha(s) afectadas. (A consulta demorou 1,6478 segundos.)  
UPDATE perfil_paises INNER JOIN estabilidade_politica ON perfil_paises.iso = estabilidade_politica.iso AND (perfil_paises.ano = 2010 AND estabilidade_politica.perc_ano_2010) SET perfil_paises.indice_estpolitica = estabilidade_politica.perc_ano_2010;  
[ Editar em linha ] [ Edita ] [ Criar código PHP ]
```

Após receber os valores, o campo *indice_estpolitica* passou a se chamar *indice_estpolitica_rank*.

Após a atualização dos campos de indicadores do Banco Mundial, os registros do campo *divisao_etnica* foram atualizados.

```
✓ 156 linha(s) afetadas. (A consulta demorou 0.8079 segundos.)
UPDATE perfil_paises INNER JOIN divisao_etnica ON perfil_paises.iso = divisao_etnica.iso AND perfil_paises.ano = divisao_etnica.ano SET perfil_paises.divisao_etnica =
divisao_etnica.indice;
[ Editar em linha ] [ Edita ] [ Criar código PHP ]
```

Em seguida, os registros da tabela *regime_politico* atualizaram os registros do campo *regime_politico* da tabela *perfil_paises*.

```
✓ 1782 linha(s) afetadas. (A consulta demorou 12,9550 segundos.)
UPDATE perfil_paises INNER JOIN regime_politico ON perfil_paises.iso = regime_politico.iso AND perfil_paises.ano = regime_politico.ano AND regime_politico.ISO != '' SET
perfil_paises.regime_politico = regime_politico.descricao_regime;
[ Editar em linha ] [ Edita ] [ Criar código PHP ]
```

Após o uso da função *join* para combinar registros entre as tabelas foi necessário tratar esses dados, como por exemplo, gerar médias, atualizar registros faltantes e classificá-los em categorias de acordo com seus valores.

Inicialmente, foi gerada a média para todos os registros utilizando como comparativo apenas a região dos países. A utilização dessa média gerada permitiu que campos numéricos vazios fossem atualizados com essa informação gerada. Assim, as médias encontradas preencheriam os registros faltantes.

Os campos utilizados para definição das médias foram os campos *indice_idh*, *indice_desemprego*, *indice_educacao* e *indice_expvida*. Os registros foram divididos em grupos por continentes e regiões, assim os resultados foram utilizados para atualizar os campos com os finais *regiao* e *continente*.

Para ilustrar os processos realizados para obtenção das médias serão utilizadas imagens referentes ao índice de IDH. Os processos foram os mesmos para todos os índices do Relatório de Desenvolvimento Humano e para cada região.

Inicialmente, foi realizado o cálculo para obter a média de cada região.

```
1 SELECT AVG(indice_idh) FROM `perfil_paises` WHERE codigo_regiao_gtd = "4" and ano = 2010 and indice_idh is NOT NULL;
2 SELECT AVG(indice_idh) FROM `perfil_paises` WHERE codigo_regiao_gtd = "4" and ano = 2011 and indice_idh is NOT NULL;
3 SELECT AVG(indice_idh) FROM `perfil_paises` WHERE codigo_regiao_gtd = "4" and ano = 2012 and indice_idh is NOT NULL;
4 SELECT AVG(indice_idh) FROM `perfil_paises` WHERE codigo_regiao_gtd = "4" and ano = 2013 and indice_idh is NOT NULL;
5 SELECT AVG(indice_idh) FROM `perfil_paises` WHERE codigo_regiao_gtd = "4" and ano = 2014 and indice_idh is NOT NULL;
6 SELECT AVG(indice_idh) FROM `perfil_paises` WHERE codigo_regiao_gtd = "4" and ano = 2015 and indice_idh is NOT NULL;
7 SELECT AVG(indice_idh) FROM `perfil_paises` WHERE codigo_regiao_gtd = "4" and ano = 2016 and indice_idh is NOT NULL;
8 SELECT AVG(indice_idh) FROM `perfil_paises` WHERE codigo_regiao_gtd = "4" and ano = 2017 and indice_idh is NOT NULL;
9 SELECT AVG(indice_idh) FROM `perfil_paises` WHERE codigo_regiao_gtd = "4" and ano = 2018 and indice_idh is NOT NULL;
```

Em seguida, o cálculo para obter a média dos dois continentes.

```
1 SELECT AVG(indice_idh) FROM `perfil_paises` WHERE nome_continente = "Africa" and ano = 2010 and indice_idh is NOT NULL;
2 SELECT AVG(indice_idh) FROM `perfil_paises` WHERE nome_continente = "Africa" and ano = 2011 and indice_idh is NOT NULL;
3 SELECT AVG(indice_idh) FROM `perfil_paises` WHERE nome_continente = "Africa" and ano = 2012 and indice_idh is NOT NULL;
4 SELECT AVG(indice_idh) FROM `perfil_paises` WHERE nome_continente = "Africa" and ano = 2013 and indice_idh is NOT NULL;
5 SELECT AVG(indice_idh) FROM `perfil_paises` WHERE nome_continente = "Africa" and ano = 2014 and indice_idh is NOT NULL;
6 SELECT AVG(indice_idh) FROM `perfil_paises` WHERE nome_continente = "Africa" and ano = 2015 and indice_idh is NOT NULL;
7 SELECT AVG(indice_idh) FROM `perfil_paises` WHERE nome_continente = "Africa" and ano = 2016 and indice_idh is NOT NULL;
8 SELECT AVG(indice_idh) FROM `perfil_paises` WHERE nome_continente = "Africa" and ano = 2017 and indice_idh is NOT NULL;
9 SELECT AVG(indice_idh) FROM `perfil_paises` WHERE nome_continente = "Africa" and ano = 2018 and indice_idh is NOT NULL;
```

```

1 SELECT AVG(indice_idh) FROM `perfil_paises` WHERE nome_continente = "Asia" and ano = 2010 and indice_idh is NOT NULL;
2 SELECT AVG(indice_idh) FROM `perfil_paises` WHERE nome_continente = "Asia" and ano = 2011 and indice_idh is NOT NULL;
3 SELECT AVG(indice_idh) FROM `perfil_paises` WHERE nome_continente = "Asia" and ano = 2012 and indice_idh is NOT NULL;
4 SELECT AVG(indice_idh) FROM `perfil_paises` WHERE nome_continente = "Asia" and ano = 2013 and indice_idh is NOT NULL;
5 SELECT AVG(indice_idh) FROM `perfil_paises` WHERE nome_continente = "Asia" and ano = 2014 and indice_idh is NOT NULL;
6 SELECT AVG(indice_idh) FROM `perfil_paises` WHERE nome_continente = "Asia" and ano = 2015 and indice_idh is NOT NULL;
7 SELECT AVG(indice_idh) FROM `perfil_paises` WHERE nome_continente = "Asia" and ano = 2016 and indice_idh is NOT NULL;
8 SELECT AVG(indice_idh) FROM `perfil_paises` WHERE nome_continente = "Asia" and ano = 2017 and indice_idh is NOT NULL;
9 SELECT AVG(indice_idh) FROM `perfil_paises` WHERE nome_continente = "Asia" and ano = 2018 and indice_idh is NOT NULL;

```

Após as médias serem encontradas, os registros de início *indice* que estavam vazios foram atualizados com os valores obtidos através dos cálculos das médias de cada região para cada índice (*indice_idh*, *indice_desemprego*, *indice_educacao* e *indice_expvida*). Como exemplo, temos o preenchimento de valores faltantes com as médias encontradas para a região onde o código de identificação é “4”. De maneira semelhante, o processo foi realizado para todas as regiões.

```

1 UPDATE perfil_paises set indice_idh = 0.815 WHERE ano = 2010 and codigo_regiao_gtd = "4" and indice_idh is NULL;
2 UPDATE perfil_paises set indice_idh = 0.822 WHERE ano = 2011 and codigo_regiao_gtd = "4" and indice_idh is NULL;
3 UPDATE perfil_paises set indice_idh = 0.828 WHERE ano = 2012 and codigo_regiao_gtd = "4" and indice_idh is NULL;
4 UPDATE perfil_paises set indice_idh = 0.834 WHERE ano = 2013 and codigo_regiao_gtd = "4" and indice_idh is NULL;
5 UPDATE perfil_paises set indice_idh = 0.839 WHERE ano = 2014 and codigo_regiao_gtd = "4" and indice_idh is NULL;
6 UPDATE perfil_paises set indice_idh = 0.843 WHERE ano = 2015 and codigo_regiao_gtd = "4" and indice_idh is NULL;
7 UPDATE perfil_paises set indice_idh = 0.846 WHERE ano = 2016 and codigo_regiao_gtd = "4" and indice_idh is NULL;
8 UPDATE perfil_paises set indice_idh = 0.849 WHERE ano = 2017 and codigo_regiao_gtd = "4" and indice_idh is NULL;
9 UPDATE perfil_paises set indice_idh = 0.853 WHERE ano = 2018 and codigo_regiao_gtd = "4" and indice_idh is NULL;

```

O processo para obtenção das médias de Indicadores de Governança do Banco Mundial foi realizado seguindo o padrão anterior. Assim, foram obtidos os valores para cada região em cada ano do período da pesquisa.

```

1 SELECT ROUND(AVG(indice_ctrlcorrupcao_rank),2) FROM `perfil_paises` WHERE codigo_regiao_gtd = "4" and ano = 2010 and indice_ctrlcorrupcao_rank is NOT NULL;
2 SELECT ROUND(AVG(indice_ctrlcorrupcao_rank),2) FROM `perfil_paises` WHERE codigo_regiao_gtd = "4" and ano = 2011 and indice_ctrlcorrupcao_rank is NOT NULL;
3 SELECT ROUND(AVG(indice_ctrlcorrupcao_rank),2) FROM `perfil_paises` WHERE codigo_regiao_gtd = "4" and ano = 2012 and indice_ctrlcorrupcao_rank is NOT NULL;
4 SELECT ROUND(AVG(indice_ctrlcorrupcao_rank),2) FROM `perfil_paises` WHERE codigo_regiao_gtd = "4" and ano = 2013 and indice_ctrlcorrupcao_rank is NOT NULL;
5 SELECT ROUND(AVG(indice_ctrlcorrupcao_rank),2) FROM `perfil_paises` WHERE codigo_regiao_gtd = "4" and ano = 2014 and indice_ctrlcorrupcao_rank is NOT NULL;
6 SELECT ROUND(AVG(indice_ctrlcorrupcao_rank),2) FROM `perfil_paises` WHERE codigo_regiao_gtd = "4" and ano = 2015 and indice_ctrlcorrupcao_rank is NOT NULL;
7 SELECT ROUND(AVG(indice_ctrlcorrupcao_rank),2) FROM `perfil_paises` WHERE codigo_regiao_gtd = "4" and ano = 2016 and indice_ctrlcorrupcao_rank is NOT NULL;
8 SELECT ROUND(AVG(indice_ctrlcorrupcao_rank),2) FROM `perfil_paises` WHERE codigo_regiao_gtd = "4" and ano = 2017 and indice_ctrlcorrupcao_rank is NOT NULL;
9 SELECT ROUND(AVG(indice_ctrlcorrupcao_rank),2) FROM `perfil_paises` WHERE codigo_regiao_gtd = "4" and ano = 2018 and indice_ctrlcorrupcao_rank is NOT NULL;

```

Em seguida, os valores foram atualizados para cada campo com registros faltantes.

Para a definição das médias dos índices de divisão étnica foram utilizados os registros do campo *divisao_etnica* de cada região.

```

1 SELECT AVG(divisao_etnica) FROM `perfil_paises` WHERE codigo_regiao_gtd = "4" and divisao_etnica is NOT NULL;
2 SELECT AVG(divisao_etnica) FROM `perfil_paises` WHERE codigo_regiao_gtd = "5" and divisao_etnica is NOT NULL;
3 SELECT AVG(divisao_etnica) FROM `perfil_paises` WHERE codigo_regiao_gtd = "6" and divisao_etnica is NOT NULL;
4 SELECT AVG(divisao_etnica) FROM `perfil_paises` WHERE codigo_regiao_gtd = "7" and divisao_etnica is NOT NULL;
5 SELECT AVG(divisao_etnica) FROM `perfil_paises` WHERE codigo_regiao_gtd = "10" and divisao_etnica is NOT NULL;
6 SELECT AVG(divisao_etnica) FROM `perfil_paises` WHERE codigo_regiao_gtd = "11" and divisao_etnica is NOT NULL;

```

De forma semelhante aos registros anteriormente atualizados, os registros faltantes do campo *divisao_etnica* foram atualizados após obter-se a média por regiões, possibilitando assim, a atualização dos registros do campo

indice_divisao_etnica. Os registros individuais de cada país foram replicados para todos os anos da pesquisa.

```

1 UPDATE perfil_paises set divisao_etnica = 0.159 WHERE ano = 2010 and codigo_regiao_gtd = "4" and divisao_etnica is NULL AND iso is NOT NULL;
2 UPDATE perfil_paises set divisao_etnica = 0.540 WHERE ano = 2010 and codigo_regiao_gtd = "5" and divisao_etnica is NULL AND iso is NOT NULL;
3 UPDATE perfil_paises set divisao_etnica = 0.545 WHERE ano = 2010 and codigo_regiao_gtd = "6" and divisao_etnica is NULL AND iso is NOT NULL;
4 UPDATE perfil_paises set divisao_etnica = 0.331 WHERE ano = 2010 and codigo_regiao_gtd = "7" and divisao_etnica is NULL AND iso is NOT NULL;
5 UPDATE perfil_paises set divisao_etnica = 0.386 WHERE ano = 2010 and codigo_regiao_gtd = "10" and divisao_etnica is NULL AND iso is NOT NULL;
6 UPDATE perfil_paises set divisao_etnica = 0.656 WHERE ano = 2010 and codigo_regiao_gtd = "11" and divisao_etnica is NULL AND iso is NOT NULL;
```

Com mais nenhum registro numérico faltante, os índices começaram a ser classificados.

Para determinar os valores a serem preenchidos nos índices do Relatório de Desenvolvimento Humano por região e por continente, foram utilizados os valores obtidos através da média:

- Os campos com resultados inferiores à média receberam o valor “*Below Average*”
- Os campos com resultados superiores à média receberam os valores “*Above Average*”
- Os campos com resultados iguais aos da média obtida receberam os valores “*Average*”

Os registros do campo *índice_idh* foram utilizados para determinar categorias como: *índice_idh_regiao* e *índice_idh_continente*. Para o campo *índice_idh_mundo* foi utilizado o registro da própria ONU.

Primeiramente os registros de classificação para os índices por região foram determinados e atualizados. Como exemplo, será o mostrado o processo de atualização dos registros onde o código da região é igual a “4”. De maneira semelhante, o processo foi realizado para todas as regiões.

```
1 UPDATE perfil_paises SET indice_idh_regiao = "Below Average" WHERE ano = 2010 and codigo_regiao_gtd = "4" and indice_idh < 0.815;
2 UPDATE perfil_paises SET indice_idh_regiao = "Average" WHERE ano = 2010 and codigo_regiao_gtd = "4" and indice_idh = 0.815;
3 UPDATE perfil_paises SET indice_idh_regiao = "Above Average" WHERE ano = 2010 and codigo_regiao_gtd = "4" and indice_idh > 0.815;
4
5 UPDATE perfil_paises SET indice_idh_regiao = "Below Average" WHERE ano = 2011 and codigo_regiao_gtd = "4" and indice_idh < 0.822;
6 UPDATE perfil_paises SET indice_idh_regiao = "Average" WHERE ano = 2011 and codigo_regiao_gtd = "4" and indice_idh = 0.822;
7 UPDATE perfil_paises SET indice_idh_regiao = "Above Average" WHERE ano = 2011 and codigo_regiao_gtd = "4" and indice_idh > 0.822;
8
9 UPDATE perfil_paises SET indice_idh_regiao = "Below Average" WHERE ano = 2012 and codigo_regiao_gtd = "4" and indice_idh < 0.828;
10 UPDATE perfil_paises SET indice_idh_regiao = "Average" WHERE ano = 2012 and codigo_regiao_gtd = "4" and indice_idh = 0.828;
11 UPDATE perfil_paises SET indice_idh_regiao = "Above Average" WHERE ano = 2012 and codigo_regiao_gtd = "4" and indice_idh > 0.828;
12
13 UPDATE perfil_paises SET indice_idh_regiao = "Below Average" WHERE ano = 2013 and codigo_regiao_gtd = "4" and indice_idh < 0.834;
14 UPDATE perfil_paises SET indice_idh_regiao = "Average" WHERE ano = 2013 and codigo_regiao_gtd = "4" and indice_idh = 0.834;
15 UPDATE perfil_paises SET indice_idh_regiao = "Above Average" WHERE ano = 2013 and codigo_regiao_gtd = "4" and indice_idh > 0.834;
16
17 UPDATE perfil_paises SET indice_idh_regiao = "Below Average" WHERE ano = 2014 and codigo_regiao_gtd = "4" and indice_idh < 0.839;
18 UPDATE perfil_paises SET indice_idh_regiao = "Average" WHERE ano = 2014 and codigo_regiao_gtd = "4" and indice_idh = 0.839;
19 UPDATE perfil_paises SET indice_idh_regiao = "Above Average" WHERE ano = 2014 and codigo_regiao_gtd = "4" and indice_idh > 0.839;
20
21 UPDATE perfil_paises SET indice_idh_regiao = "Below Average" WHERE ano = 2015 and codigo_regiao_gtd = "4" and indice_idh < 0.843;
22 UPDATE perfil_paises SET indice_idh_regiao = "Average" WHERE ano = 2015 and codigo_regiao_gtd = "4" and indice_idh = 0.843;
23 UPDATE perfil_paises SET indice_idh_regiao = "Above Average" WHERE ano = 2015 and codigo_regiao_gtd = "4" and indice_idh > 0.843;
24
25 UPDATE perfil_paises SET indice_idh_regiao = "Below Average" WHERE ano = 2016 and codigo_regiao_gtd = "4" and indice_idh < 0.846;
26 UPDATE perfil_paises SET indice_idh_regiao = "Average" WHERE ano = 2016 and codigo_regiao_gtd = "4" and indice_idh = 0.846;
27 UPDATE perfil_paises SET indice_idh_regiao = "Above Average" WHERE ano = 2016 and codigo_regiao_gtd = "4" and indice_idh > 0.846;
28
29 UPDATE perfil_paises SET indice_idh_regiao = "Below Average" WHERE ano = 2017 and codigo_regiao_gtd = "4" and indice_idh < 0.849;
30 UPDATE perfil_paises SET indice_idh_regiao = "Average" WHERE ano = 2017 and codigo_regiao_gtd = "4" and indice_idh = 0.849;
31 UPDATE perfil_paises SET indice_idh_regiao = "Above Average" WHERE ano = 2017 and codigo_regiao_gtd = "4" and indice_idh > 0.849;
32
33 UPDATE perfil_paises SET indice_idh_regiao = "Below Average" WHERE ano = 2018 and codigo_regiao_gtd = "4" and indice_idh < 0.853;
34 UPDATE perfil_paises SET indice_idh_regiao = "Average" WHERE ano = 2018 and codigo_regiao_gtd = "4" and indice_idh = 0.853;
35 UPDATE perfil_paises SET indice_idh_regiao = "Above Average" WHERE ano = 2018 and codigo_regiao_gtd = "4" and indice_idh > 0.853;
```

Em seguida, os índices de classificação para os continentes.

Primeiro para o continente africano.

E depois para o continente asiático.

```

1 UPDATE perfil_paises SET indice_idh_continente = "Below Average" WHERE ano = 2010 and nome_continente = "Asia" and indice_idh < 0.704;
2 UPDATE perfil_paises SET indice_idh_continente = "Average" WHERE ano = 2010 and nome_continente = "Asia" and indice_idh = 0.704;
3 UPDATE perfil_paises SET indice_idh_continente = "Above Average" WHERE ano = 2010 and nome_continente = "Asia" and indice_idh > 0.704;
4
5 UPDATE perfil_paises SET indice_idh_continente = "Below Average" WHERE ano = 2011 and nome_continente = "Asia" and indice_idh < 0.710;
6 UPDATE perfil_paises SET indice_idh_continente = "Average" WHERE ano = 2011 and nome_continente = "Asia" and indice_idh = 0.710;
7 UPDATE perfil_paises SET indice_idh_continente = "Above Average" WHERE ano = 2011 and nome_continente = "Asia" and indice_idh > 0.710;
8
9 UPDATE perfil_paises SET indice_idh_continente = "Below Average" WHERE ano = 2012 and nome_continente = "Asia" and indice_idh < 0.716;
10 UPDATE perfil_paises SET indice_idh_continente = "Average" WHERE ano = 2012 and nome_continente = "Asia" and indice_idh = 0.716;
11 UPDATE perfil_paises SET indice_idh_continente = "Above Average" WHERE ano = 2012 and nome_continente = "Asia" and indice_idh > 0.716;
12
13 UPDATE perfil_paises SET indice_idh_continente = "Below Average" WHERE ano = 2013 and nome_continente = "Asia" and indice_idh < 0.719;
14 UPDATE perfil_paises SET indice_idh_continente = "Average" WHERE ano = 2013 and nome_continente = "Asia" and indice_idh = 0.719;
15 UPDATE perfil_paises SET indice_idh_continente = "Above Average" WHERE ano = 2013 and nome_continente = "Asia" and indice_idh > 0.719;
16
17 UPDATE perfil_paises SET indice_idh_continente = "Below Average" WHERE ano = 2014 and nome_continente = "Asia" and indice_idh < 0.723;
18 UPDATE perfil_paises SET indice_idh_continente = "Average" WHERE ano = 2014 and nome_continente = "Asia" and indice_idh = 0.723;
19 UPDATE perfil_paises SET indice_idh_continente = "Above Average" WHERE ano = 2014 and nome_continente = "Asia" and indice_idh > 0.723;
20
21 UPDATE perfil_paises SET indice_idh_continente = "Below Average" WHERE ano = 2015 and nome_continente = "Asia" and indice_idh < 0.727;
22 UPDATE perfil_paises SET indice_idh_continente = "Average" WHERE ano = 2015 and nome_continente = "Asia" and indice_idh = 0.727;
23 UPDATE perfil_paises SET indice_idh_continente = "Above Average" WHERE ano = 2015 and nome_continente = "Asia" and indice_idh > 0.727;
24
25 UPDATE perfil_paises SET indice_idh_continente = "Below Average" WHERE ano = 2016 and nome_continente = "Asia" and indice_idh < 0.730;
26 UPDATE perfil_paises SET indice_idh_continente = "Average" WHERE ano = 2016 and nome_continente = "Asia" and indice_idh = 0.730;
27 UPDATE perfil_paises SET indice_idh_continente = "Above Average" WHERE ano = 2016 and nome_continente = "Asia" and indice_idh > 0.730;
28
29 UPDATE perfil_paises SET indice_idh_continente = "Below Average" WHERE ano = 2017 and nome_continente = "Asia" and indice_idh < 0.734;
30 UPDATE perfil_paises SET indice_idh_continente = "Average" WHERE ano = 2017 and nome_continente = "Asia" and indice_idh = 0.734;
31 UPDATE perfil_paises SET indice_idh_continente = "Above Average" WHERE ano = 2017 and nome_continente = "Asia" and indice_idh > 0.734;
32
33 UPDATE perfil_paises SET indice_idh_continente = "Below Average" WHERE ano = 2018 and nome_continente = "Asia" and indice_idh < 0.737;
34 UPDATE perfil_paises SET indice_idh_continente = "Average" WHERE ano = 2018 and nome_continente = "Asia" and indice_idh = 0.737;
35 UPDATE perfil_paises SET indice_idh_continente = "Above Average" WHERE ano = 2018 and nome_continente = "Asia" and indice_idh > 0.737;

```

Para determinar os valores do campo *faixa_idh* foram utilizados os valores determinados pela ONU:

- *Very High* para índices iguais ou acima de 0.800;
- *High* para índices iguais ou acima de 0.700 e inferiores a 0.800;
- *Medium* para índices entre 0.550 e 0.699;
- *Low* para índices abaixo de 0.550.

```

1 UPDATE perfil_paises SET faixa_idh = "Very High" WHERE indice_idh >= 0.800;
2 UPDATE perfil_paises SET faixa_idh = "High" WHERE indice_idh BETWEEN 0.700 AND 0.799;
3 UPDATE perfil_paises SET faixa_idh = "Medium" WHERE indice_idh BETWEEN 0.550 AND 0.699;
4 UPDATE perfil_paises SET faixa_idh = "Low" WHERE indice_idh <= 0.549;

```

Da mesma maneira, os valores do campo *indice_desemprego* foram utilizados para determinar as categorias *indice_desemprego_regiao* e *indice_desemprego_continente*. Para o campo *indice_desemprego_mundo* foi utilizado o registro da própria ONU.

Assim como para os registros anteriores, os valores do campo *indice_educacao* foram utilizados para determinar as categorias *indice_educacao_regiao* e *indice_educacao_continente*. Para o campo *indice_educacao_mundo* foi utilizado o registro da própria ONU.

Os valores do campo *indice_expvida* foram utilizados para determinar categorias como: *indice_expvida_regiao* e *indice_expvida_continente*. Para o campo *indice_expvida_mundo* foi utilizado o registro da própria ONU.

Em seguida, os registros do campo *indice_ctrlcorrupcao* foram atualizados após obter-se a média dos registros do campo *indice_ctrlcorrupcao_rank*. Os dados variam entre 0% e 100%. Nesse caso, valores abaixo da média (50%) receberam o índice “Low Index”, enquanto os valores iguais ou acima da média receberam o valor “High Index”.

```
1 UPDATE perfil_paises SET indice_ctrlcorrupcao = "Low Rank" WHERE indice_ctrlcorrupcao_rank < 50;
2 UPDATE perfil_paises SET indice_ctrlcorrupcao = "High Rank" WHERE indice_ctrlcorrupcao_rank >= 50;
```

Depois, os registros do campo *indice_efetividadegov* foram atualizados.

```
1 UPDATE perfil_paises SET indice_efetividadegov = "Low Rank" WHERE indice_efetividadegov_rank < 50;
2 UPDATE perfil_paises SET indice_efetividadegov = "High Rank" WHERE indice_efetividadegov_rank >= 50;
```

E por fim, os registros do campo *indice_estpolitica* foram atualizados.

```
1 UPDATE perfil_paises SET indice_estpolitica = "Low Rank" WHERE indice_estpolitica_rank < 50;
2 UPDATE perfil_paises SET indice_estpolitica = "High Rank" WHERE indice_estpolitica_rank >= 50;
```

Para a classificação dos índices de divisão étnica foi utilizada uma média de todos os territórios pesquisados que possuíam registros. Registros menores do que as médias receberam o valor “Low Index”, enquanto registros iguais ou acima das médias receberam o valor “High Index”.

```
1 UPDATE perfil_paises SET indice_divisao_etnica = "High Index" WHERE divisao_etnica >= 0.454;
2 UPDATE perfil_paises SET indice_divisao_etnica = "Low Index" WHERE divisao_etnica < 0.454;
```

Para os registros do campo *regime_politico* não foi possível realizar cálculos de média ou fazer uma aproximação por região. Os registros faltantes receberam o valor “*other classification*” (outra classificação). A escolha por essa alteração se dá pela dificuldade em classificar sistemas políticos de territórios que não são reconhecidas ou ainda são consideradas colônias.

Ao final das alterações é possível visualizar as informações da tabela.

| Field | Type | Null | Key | Default | Extra |
|------------------------------|---------------|------|-----|---------|----------------|
| id | int(6) | NO | PRI | NULL | auto_increment |
| nome_continente | varchar(10) | YES | | NULL | |
| codigo_regiao_gtd | int(2) | YES | | NULL | |
| nome_regiao | varchar(30) | YES | | NULL | |
| representacao_pais | varchar(20) | YES | | NULL | |
| nome_pais | varchar(96) | YES | | NULL | |
| iso | varchar(2) | YES | | NULL | |
| ano | int(4) | YES | | NULL | |
| regime_politico | varchar(57) | YES | | NULL | |
| divisao_etnica | decimal(10,3) | YES | | NULL | |
| indice_divisao_etnica | varchar(20) | YES | | NULL | |
| religiao_predominante | varchar(20) | YES | | NULL | |
| religiao_secundaria | varchar(20) | YES | | NULL | |
| indice_desemprego | decimal(10,1) | YES | | NULL | |
| indice_desemprego_regiao | varchar(20) | YES | | NULL | |
| indice_desemprego_continente | varchar(20) | YES | | NULL | |
| indice_desemprego_mundo | varchar(20) | YES | | NULL | |
| indice_educacao | decimal(10,3) | YES | | NULL | |
| indice_educacao_regiao | varchar(20) | YES | | NULL | |
| indice_educacao_continente | varchar(20) | YES | | NULL | |
| indice_educacao_mundo | varchar(20) | YES | | NULL | |
| indice_expvida | decimal(10,1) | YES | | NULL | |
| indice_expvida_regiao | varchar(20) | YES | | NULL | |
| indice_expvida_continente | varchar(20) | YES | | NULL | |
| indice_expvida_mundo | varchar(20) | YES | | NULL | |
| indice_idh | decimal(10,3) | YES | | NULL | |
| faixa_idh | varchar(20) | YES | | NULL | |
| indice_idh_regiao | varchar(20) | YES | | NULL | |
| indice_idh_continente | varchar(20) | YES | | NULL | |
| indice_idh_mundo | varchar(20) | YES | | NULL | |
| indice_ctrlcorrupcao_rank | decimal(10,2) | YES | | NULL | |
| indice_ctrlcorrupcao | varchar(20) | YES | | NULL | |
| indice_efetividadegov_rank | decimal(10,2) | YES | | NULL | |
| indice_efetividadegov | varchar(20) | YES | | NULL | |
| indice_estpolitica_rank | decimal(10,2) | YES | | NULL | |
| indice_estpolitica | varchar(20) | YES | | NULL | |

6. Análise e exploração dos dados

A seção de análise e exploração dos dados foi desenvolvida em Python e através do programa Microsoft BI após o processamento e tratamento dos dados.

Inicialmente, busca-se através dessa seção expor de forma natural números e dados estatísticos sobre o material estudado. Para maior compreensão, optou-se por dividir essa seção em três partes.

6.1. Exploração superficial sobre a quantidade de territórios estudados

Após o carregamento da tabela *perfil_paises* e a descrição de parte de sua estrutura, como a quantidade de registros, nome e número de colunas, iniciou-se o processo de análise dos dados.

Primeiramente, é possível observar que entre os dois continentes estudados, foram reunidas informações para 108 territórios, sendo 57 deles no continente africano e os outros 51 no continente asiático.

```
num_territorios_continente = analise.groupby(['nome_continente'])['id'].count()
print("Número de territórios representados na pesquisa, por continente: " + str(num_territorios_continente))

Número de territórios representados na pesquisa, por continente: nome_continente
África      57
Ásia        51
Name: id, dtype: int64
```

Os territórios estão divididos em seis regiões de acordo com as divisões adotadas pelo GTD.

| Código | Região adotada pelo GTD | Termo correspondente em Português |
|--------|----------------------------|--|
| 7 | Central Asia | Região Central da Ásia |
| 4 | East Asia | Região Leste da Ásia |
| 10 | Middle East & North Africa | Região que comprehende o Oriente Médio e o Norte da África |
| 6 | South Asia | Região Sul da Ásia |
| 5 | Southeast Asia | Região Sudeste da Ásia |
| 11 | Sub-Saharan Africa | Região da África Subsaariana |

É possível observar que há uma grande concentração de territórios africanos em uma única região, na África subsaariana, enquanto os territórios do continente asiático estão divididos em outras cinco regiões, inclusive juntos à região norte do continente africano.

```
num_territorios_regiao = analise.groupby(['nome_regiao'])['id'].count()
print("Número de territórios representados na pesquisa, por região: " + str(num_territorios_regiao))

Número de territórios representados na pesquisa, por região: nome_regiao
Central Asia           8
East Asia              8
Middle East & North Africa 21
South Asia             9
Southeast Asia         11
Sub-Saharan Africa    51
Name: id, dtype: int64
```

Para continuidade do estudo, os territórios foram classificados quanto a sua representatividade na pesquisa, sendo divididos em três categorias: Não Representado, Representado e Vítima. A seguir, a explicação para cada categoria:

- Não representado: territórios onde foram reunidas informações políticas e sociais, mas que não possuem registros sobre eventos terroristas.
- Representado: territórios onde foram reunidas informações políticas e sociais, e estão representados na pesquisa contendo registros de eventos terroristas.
- Vítimas: territórios onde foram reunidas informações políticas e sociais, porém estão representados apenas como “vítimas” ou “alvo” em registros de ataques terroristas.

```
territoriosRepresentados = analise.groupby(['representacao_pais'])['id'].count()
print("Número de territórios representados na pesquisa, que possuem ou não registros de eventos terroristas: " + str(territoriosRepresentados))

Nº de territórios representados na pesquisa, que possuem ou não registros de eventos terroristas: representacao_pais
Não Representado      5
Representado          101
Vítima                2
Name: id, dtype: int64
```

A discriminação dessas categorias foi feita apenas para observação dos registros, já que não houve interferência nos resultados.

6.2. Exploração específica sobre os territórios estudados

Nesta seção busca-se através de uma análise mais específica sobre os territórios investigar e dispor parte das informações políticas e sociais que foram utilizadas para desenvolvimento de parte do estudo realizado nesta pesquisa. Essas informações foram determinantes para o desenvolvimento de novos modelos de *machine learning* a fim de comparar e relacionar dados político-sociais e culturais com a intenção de mapear áreas propensas a ataques terroristas de acordo com indicadores sociais e de desenvolvimento humano.

Uma análise comparativa entre os territórios que mais possuem registros de eventos terroristas em cada região foi desenvolvida a fim de expor os dados reunidos para cada um:

| Região | Território | Registros |
|----------------------------|-------------|-----------|
| East Asia | China | 87 |
| Southeast Asia | Filipinas | 4.501 |
| South Asia | Afeganistão | 12.434 |
| Central Asia | Georgia | 24 |
| Middle East & North Africa | Iraque | 20.766 |
| Sub-Saharan Africa | Nigéria | 4.225 |

Dessa forma utilizaremos o *dashboard* desenvolvido no Microsoft Power BI para fazer uma análise sobre cada um dos territórios mencionados acima.



Philippines



| Ano | IDH | Desemprego (%) | Educação | Expect. de Vida |
|------|-------|----------------|----------|-----------------|
| 2010 | 0,671 | 3,6 | 0,62 | 69,8 |
| 2011 | 0,676 | 3,6 | 0,63 | 70,0 |
| 2012 | 0,684 | 3,5 | 0,64 | 70,1 |
| 2013 | 0,691 | 3,5 | 0,66 | 70,3 |
| 2014 | 0,696 | 3,6 | 0,66 | 70,5 |
| 2015 | 0,701 | 3,1 | 0,67 | 70,6 |
| 2016 | 0,704 | 2,7 | 0,66 | 70,8 |
| 2017 | 0,708 | 2,6 | 0,67 | 71,0 |
| 2018 | 0,711 | 2,3 | 0,67 | 71,1 |



Religião Predominante
Christians

Religião Secundária
Muslims

Índice Divisão Étnica Valor
High Index 0,81

Afghanistan



| Ano | IDH | Desemprego (%) | Educação | Expect. de Vida |
|------|-------|----------------|----------|-----------------|
| 2010 | 0,472 | 11,5 | 0,37 | 61,0 |
| 2011 | 0,477 | 11,5 | 0,37 | 61,6 |
| 2012 | 0,489 | 11,5 | 0,39 | 62,1 |
| 2013 | 0,496 | 11,5 | 0,40 | 62,5 |
| 2014 | 0,500 | 11,4 | 0,40 | 63,0 |
| 2015 | 0,500 | 11,4 | 0,41 | 63,4 |
| 2016 | 0,502 | 11,3 | 0,41 | 63,8 |
| 2017 | 0,506 | 11,2 | 0,41 | 64,1 |
| 2018 | 0,509 | 11,1 | 0,41 | 64,5 |



Religião Predominante
Muslims

Religião Secundária
Christians

Índice Divisão Étnica Valor
High Index 0,76

Georgia

| Ano | IDH | Desemprego (%) | Educação | Expect. de Vida |
|------|-------|----------------|----------|-----------------|
| 2010 | 0,751 | 20,2 | 0,78 | 71,5 |
| 2011 | 0,757 | 19,6 | 0,78 | 71,8 |
| 2012 | 0,767 | 19,7 | 0,79 | 72,1 |
| 2013 | 0,775 | 19,4 | 0,80 | 72,4 |
| 2014 | 0,783 | 17,4 | 0,82 | 72,7 |
| 2015 | 0,790 | 16,5 | 0,83 | 73,0 |
| 2016 | 0,792 | 16,6 | 0,84 | 73,2 |
| 2017 | 0,799 | 13,9 | 0,85 | 73,4 |
| 2018 | 0,805 | 13,8 | 0,85 | 73,6 |

Religião Predominante
Christians

Religião Secundária
Muslims

Índice Divisão Étnica Valor
Low Index 0,39

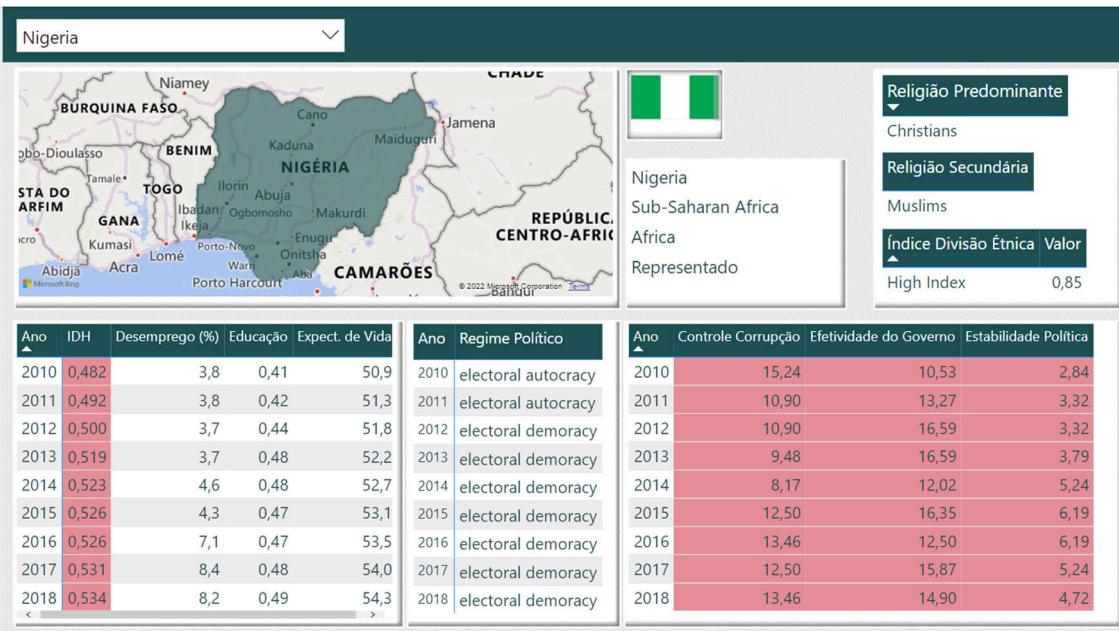
Iraq

| Ano | IDH | Desemprego (%) | Educação | Expect. de Vida |
|------|-------|----------------|----------|-----------------|
| 2010 | 0,636 | 8,4 | 0,50 | 68,6 |
| 2011 | 0,642 | 8,2 | 0,51 | 68,8 |
| 2012 | 0,646 | 8,0 | 0,51 | 69,1 |
| 2013 | 0,646 | 9,3 | 0,50 | 69,4 |
| 2014 | 0,645 | 10,6 | 0,50 | 69,7 |
| 2015 | 0,649 | 10,7 | 0,51 | 69,9 |
| 2016 | 0,656 | 10,8 | 0,51 | 70,1 |
| 2017 | 0,667 | 13,0 | 0,54 | 70,3 |
| 2018 | 0,671 | 12,9 | 0,55 | 70,5 |

Religião Predominante
Muslims

Religião Secundária
Christians

Índice Divisão Étnica Valor
Low Index 0,44



Analisando o primeiro critério para classificação de um evento como um ataque terrorista, é possível observar que informações políticas, culturais, sociais e religiosas são extremamente importantes e determinantes para este estudo.

Ao explorarmos as informações dos 6 países citados, podemos observar que os territórios localizados nas regiões Central e Leste da Ásia e do Oriente Médio e Norte da África possuem menores índices de fracionação étnica, sendo a China com o menor índice e o Iraque com um índice de divisão étnica próximo a média determinada pelo Índice Histórico de Fracionação Étnica. Ao contrário dos outros 3 países, a Nigéria na região da África subsaariana, as Filipinas na região Sudeste da África e o Afeganistão na região Sul da Ásia possuem índices de fracionação étnica muito elevados.

Ao analisarmos isoladamente a China e os territórios em sua região, como Japão, Hong Kong e Taiwan é possível relacionar o baixo número de registros de ataques terroristas aos índices de fracionação étnica extremamente abaixo da média.

Os dados referentes às religiões predominantes e secundárias nos territórios são bem variados. Esses registros são muito importantes para entendermos como as diferenças étnicas e sociais podem determinar áreas de conflitos. Dos seis países apresentados, três possuem como religião predominante o Cristianismo e como religião secundária o Islamismo. São eles: Geórgia, Filipinas e Nigéria. Essas informações podem estar relacionadas muito em virtude de suas colonizações e territórios próximos. Dos dois países que possuem como religião predominante o

Islamismo, é o Cristianismo a religião secundária. Sendo ambos os territórios, o Afeganistão e o Iraque os que possuem também os maiores registros de eventos terroristas no período estudado. Por fim, a China, país do Leste da Ásia que possui o menor índice de fracionalização étnica entre os territórios apresentados, possui em seu extenso território diversas religiões populares e o Budismo como sua religião secundária.

Dentre os seis países explorados durante essa seção, quatro deles passaram por regimes autoritários durante o período estudado: Afeganistão, China, Iraque e Nigéria. Tal regime é caracterizado pela centralização do poder em uma única personalidade, além da repressão às liberdades individuais e em casos extremos, perseguições a opositores. Dessa forma, não só apoiadores do Estado, mas também grupos opositores a um governo já estabelecido utilizam de protestos e atos não pacíficos para realização de eventos dessa natureza. É possível mencionar que além das motivações políticas para caracterização de um ato como sendo um ataque terrorista, diversos ataques não direcionados ao Estado, mas tendo a população como alvo, tem como objetivo transmitir alguma mensagem para um público maior, e muitas vezes para as autoridades.

Especialistas sociais entendem que a atuação do Estado, tanto em diretrizes de governo quanto na forma como isso se reflete na população, são determinantes para impulsionar conflitos, protestos e atos extremos como eventos terroristas. Dessa maneira, foram reunidos dados indicadores de governança e índices de desenvolvimento social dos territórios estudados.

Índices como expectativa de vida refletem as condições de vida e segurança da população, assim como índices de educação e desemprego podem retratar a desigualdade de oportunidades e diferenças entre as classes sociais de um país. A importância da análise e consideração desses dados para a pesquisa se dá pela forma como informações dessa natureza refletem a influência de diferentes condições para uma população. Condições mínimas afetam diretamente a população e podem servir de estímulos para revoltas e atos violentos.

Analisando os Indicadores de Governança do Banco Mundial, que variam entre 0% e 100%, podemos observar que 3 países: Afeganistão, Iraque e Nigéria obtêm baixos números indicadores nas três categorias: Controle de Corrupção, Efetividade do Governo e Estabilidade Política e Controle da Violência. Já os territórios do Leste e Sudeste asiático, China e Filipinas respectivamente, possuem índices positivos de

Efetividade do Governo, porém tendo a China com níveis próximos a 50% na categoria Controle de Corrupção, enquanto as Filipinas possuem números mais baixos nas outras duas categorias. A Geórgia, antigo território das Repúblicas da União Soviética, no entanto, possui indicadores muito positivos nas duas primeiras categorias e pequenas oscilações na categoria Estabilidade Política e Controle da Violência.

Os índices de desenvolvimento humano da ONU foram analisados a partir de 4 categorias: índice de desenvolvimento humano, percentual de desemprego, índice de educação e expectativa de vida.

Durante o período estudado todos os países, com exceção do Iraque, tiveram uma diminuição no percentual de desemprego. Ainda assim, durante o período a Geórgia manteve-se sempre no topo da lista no quesito, enquanto as Filipinas mantiveram baixos níveis no índice de desemprego em todo o período.

O índice de educação é apresentado pela média de anos de escolaridade de adultos acima de 25 anos e a expectativa de anos escolares para crianças que estão ingressando na escola. Durante o período estudado, países como o Iraque e a Nigéria passaram por uma oscilação nos índices de educação, onde apresentaram uma queda em certos anos até uma nova melhora. A Geórgia obteve em todos os anos os melhores resultados para o índice, sempre com um aumento. Enquanto China e Filipinas tiveram aumentos gradativos e com resultados semelhantes, no Afeganistão os resultados permaneceram os mais baixos durante todo o período, mesmo apresentando uma melhora ao longo dos anos.

Durante o período estudado, apenas a Nigéria apresentou uma queda na expectativa de vida. Embora o crescimento não tenha sido tão acentuado nos outros territórios, o país localizado na região da África subsaariana não apresentou melhorias em seu resultado. Em contrapartida, o Afeganistão, mesmo com um índice de expectativa de vida relativamente baixo, apresentou um aumento em seus números. Enquanto isso, China e Geórgia apresentaram índices acima dos 70 anos desde o primeiro ano da pesquisa.

O Índice de Desenvolvimento Humano (IDH) é apresentado em quatro níveis ou faixas: Baixo, Médio, Alto e Muito Alto. Os territórios apresentaram um IDH bem distinto entre si. Enquanto a China e a Geórgia estiveram na faixa de Alto índice por praticamente todo o período (A Geórgia atingiu o nível Muito Alto no último ano da pesquisa), as Filipinas atingiram tal feito em 2015. Já o Afeganistão e a Nigéria não

passaram da faixa de índice Baixo, sendo o território localizado no Sul da Ásia com o menor índice entre os seis.

A apresentação dos resultados dessa sessão não representa todos os territórios de cada categoria, foi utilizada para uma exploração dos dados de todos os territórios estudados (representados ou não), onde é possível analisá-los individualmente através do Power BI.

6.3. Exploração dos registros de ataques terroristas

Após o carregamento e análise sobre a estrutura da tabela `terrorismo_africa_asia`, onde estão os registros de eventos terroristas estudados durante esta pesquisa, foram iniciados os processos de análise e exploração dos dados.

Anteriormente ao processo de exploração dos dados foi realizada uma checagem de valores nulos para os campos do dataset. Após encontrados, os valores nulos receberam rótulos de acordo com a sua especificação para futuros agrupamentos e somas de registros. Dessa forma, os seguintes valores foram atualizados:

- Registros nulos no campo `cidade` receberam o valor “Not Specified”;
- Registros nulos no campo `subtipo_vitima` receberam o valor “Unknown”;
- Registros nulos no campo `nac_alvo` receberam o valor “Not Specified”.

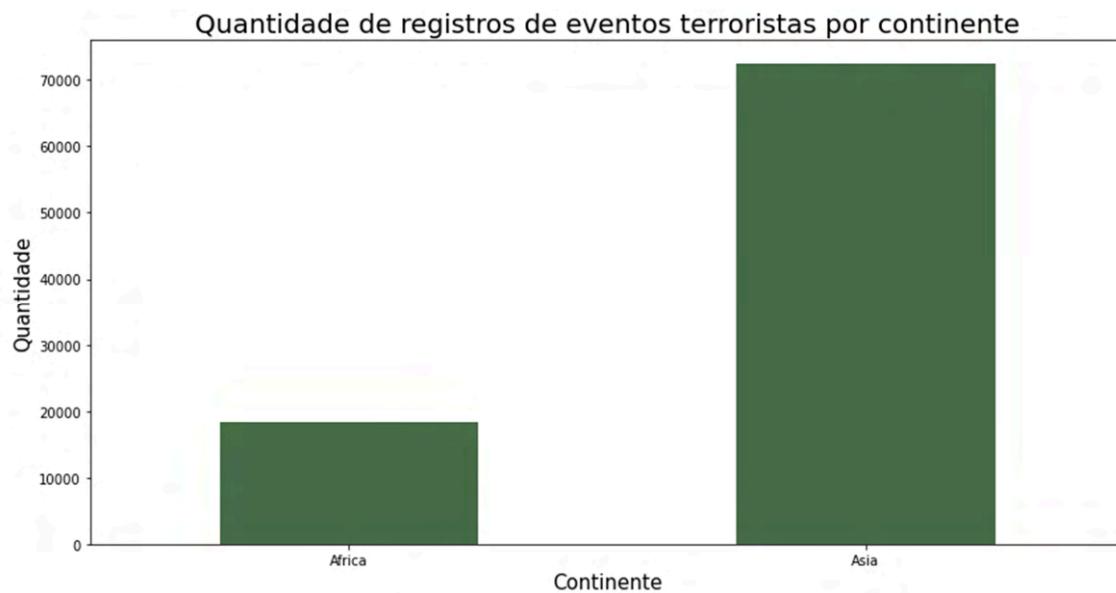
```
analise_terrorismo.update(analise_terrorismo['cidade'].fillna('Not Specified'))
analise_terrorismo.update(analise_terrorismo['subtipo_vitima'].fillna('Unknown'))
analise_terrorismo.update(analise_terrorismo['nac_alvo'].fillna('Not Specified'))
```

Inicialmente foi feita uma comparação entre a quantidade de registros de eventos terroristas entre os dois continentes durante o período pesquisado. Embora a quantidade de territórios estudados para cada continente seja quase a mesma, é possível observar uma grande diferença no número de registros de eventos terroristas entre eles, já que os números de registros no continente asiático somam quase 4 vezes o número de registros no continente africano.

```
num_registros_continente = analise_terrorismo.groupby([analise_terrorismo['nome_continente']])['id'].count()
print("Número de registros de eventos terroristas por continente: " + str(num_registros_continente))

Número de registros de eventos terroristas por continente: nome_continente
África    18377
Ásia      72378
Name: id, dtype: int64
```

Através da visualização por meio de um gráfico é possível perceber de forma mais apropriada a desproporção entre os dois continentes.



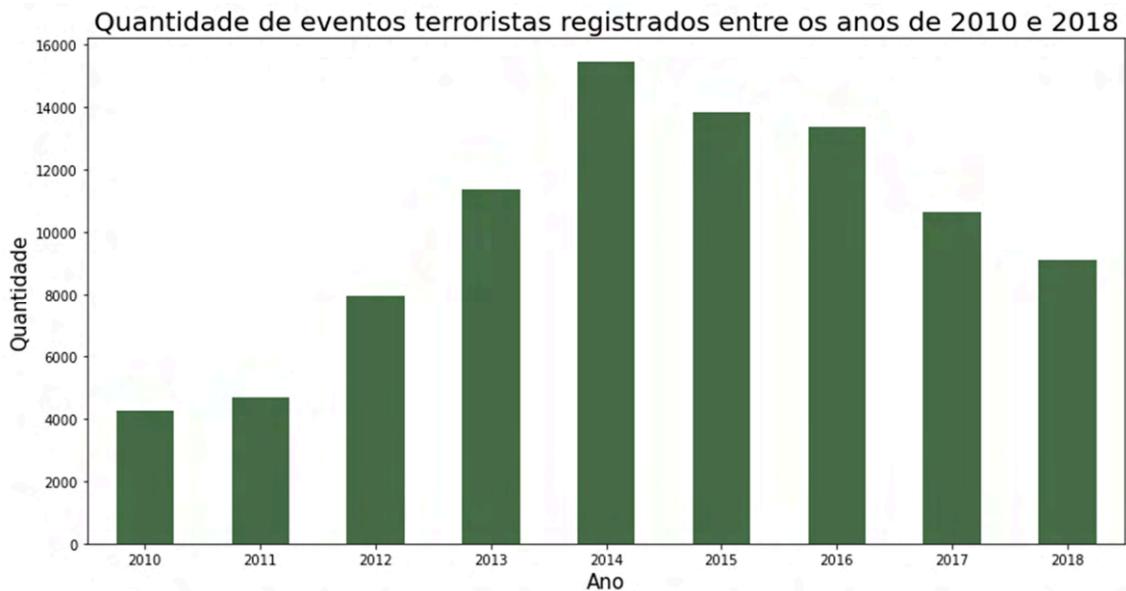
A partir de uma análise específica sobre a quantidade de registros por ano, podemos constatar que no ano de 2014 o número de registros teve seu auge, sendo registrados 15.448 eventos terroristas ao total. Já no ano de 2010, onde inicia-se a pesquisa, foram registrados 4.258 eventos, o menor número registrado durante o período pesquisado.

```
registros_anuais = analise_terrorismo.groupby(['ano'])['id'].count()
print("Número de eventos terroristas registrados entre os anos de 2010 e 2018: " + str(registros_anuais))

Número de eventos terroristas registrados entre os anos de 2010 e 2018: ano
2010    4258
2011    4664
2012    7979
2013   11393
2014  15448
2015  13861
2016  13381
2017  10652
2018   9119
Name: id, dtype: int64
```

Um dos motivos para o grande número de registros de eventos terroristas no ano de 2014 se dá pela intensa atividade do grupo terrorista do Estado Islâmico ISIS,

sobretudo no Iraque e na Síria.



É possível observar que embora os registros de eventos terroristas tenham caído entre os anos de 2015 e 2018, apenas no último ano o número de registros foi abaixo de 10.000 ocorrências.

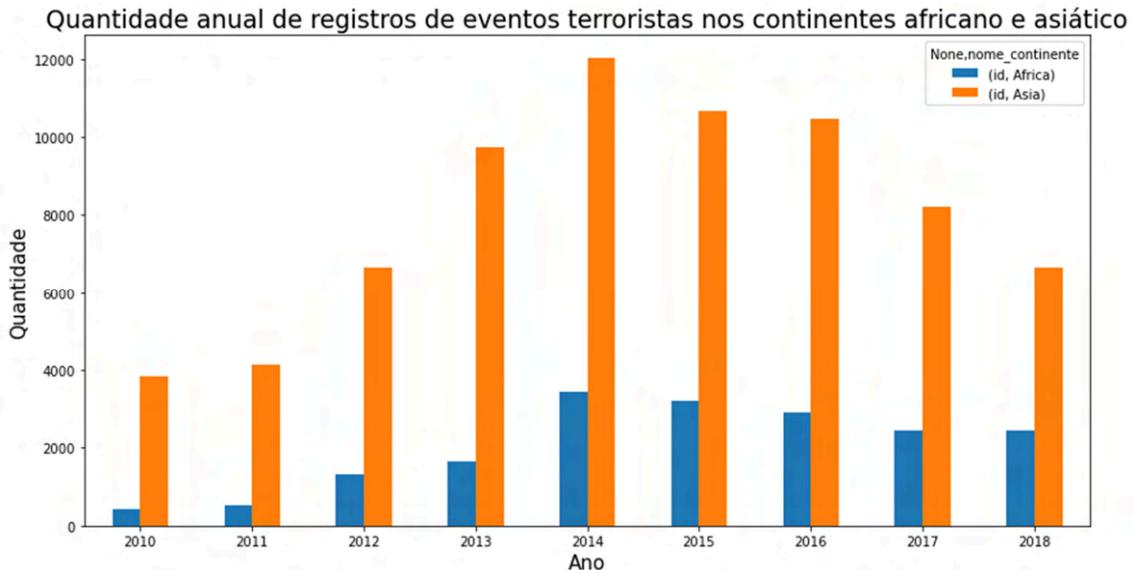
Ao analisarmos especificamente o período de tempo pesquisado e os continentes estudados, ambos também tiveram o seu auge de registros de eventos terroristas no ano de 2014, onde 3.439 foram registrados na África e 12.009 na Ásia. Assim como na análise anual geral, o ano de 2010 também foi o ano com os menores registros de eventos nos continentes africano e asiático, que contabilizaram 433 e 3.825 registros, respectivamente.

```

num_registros_anuais_continente = analise_terrorismo.groupby(['ano','analise_terrorismo['nome_continente']'])['
print("Quantidade de registros de eventos terroristas nos continentes africano e asiático por ano: " + str(num_registros_anuais_c
+ 
Quantidade de registros de eventos terroristas nos continentes africano e asiático por ano: ano  nome_continente
2010  Africa      433
      Asia       3825
2011  Africa      533
      Asia       4131
2012  Africa     1317
      Asia      6662
2013  Africa     1664
      Asia      9729
2014  Africa     3439
      Asia     12009
2015  Africa     3196
      Asia     10665
2016  Africa     2901
      Asia     10480
2017  Africa     2434
      Asia      8218
2018  Africa     2460
      Asia      6659
Name: id, dtype: int64

```

No continente asiático, entre 2014 e 2016 cada ano do período registrou mais de 10.000 eventos terroristas, sendo que, no ano anterior, o número de eventos foi próximo a essa marca.



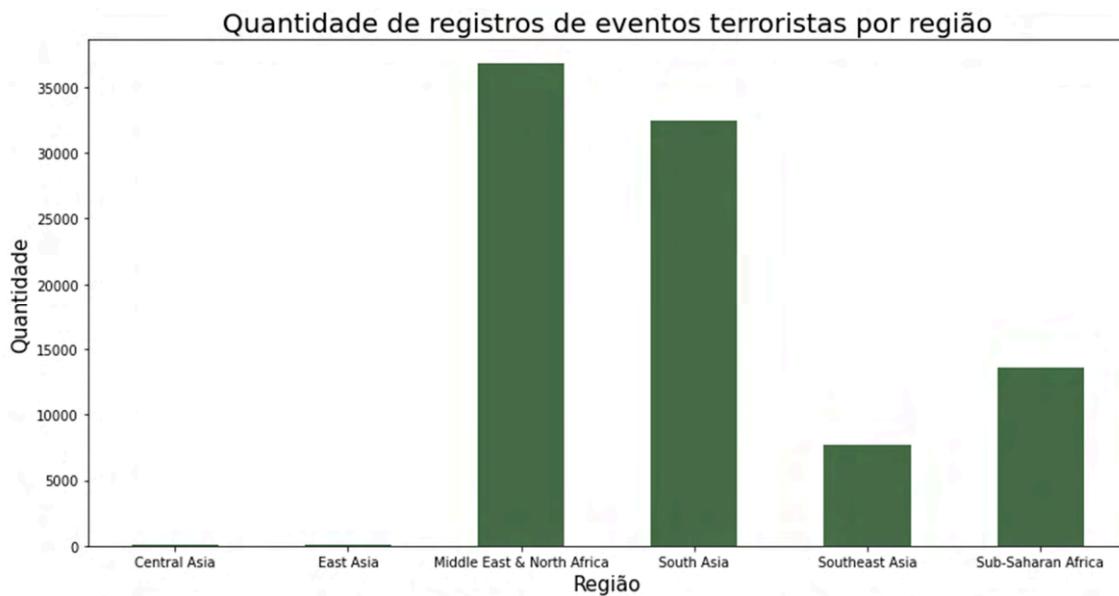
Os intensos e recorrentes conflitos nas regiões da Ásia durante o século levaram a divisões políticas e sociais em diversos territórios. Seja por motivações políticas, econômicas, sociais ou culturais, as consequências desses conflitos continuaram a motivar novos combates.

Através de uma análise geral sobre a quantidade de registros de eventos terroristas entre as regiões, é possível observar que a região do Oriente Médio e Norte da África possui o maior número de registros, seguidos pelas regiões sul da Ásia e dos territórios localizados na África subsaariana. Em contrapartida, as regiões sudeste, leste e central da Ásia registram as menores quantidades de eventos.

```
num_registros_regiao = analise_terrorismo.groupby(['nome_regiao'])['id'].count()
print("Número de registros de eventos terroristas por região: " + str(num_registros_regiao))

Número de registros de eventos terroristas por região: nome_regiao
Central Asia           88
East Asia              116
Middle East & North Africa 36763
South Asia             32477
Southeast Asia          7735
Sub-Saharan Africa      13576
Name: id, dtype: int64
```

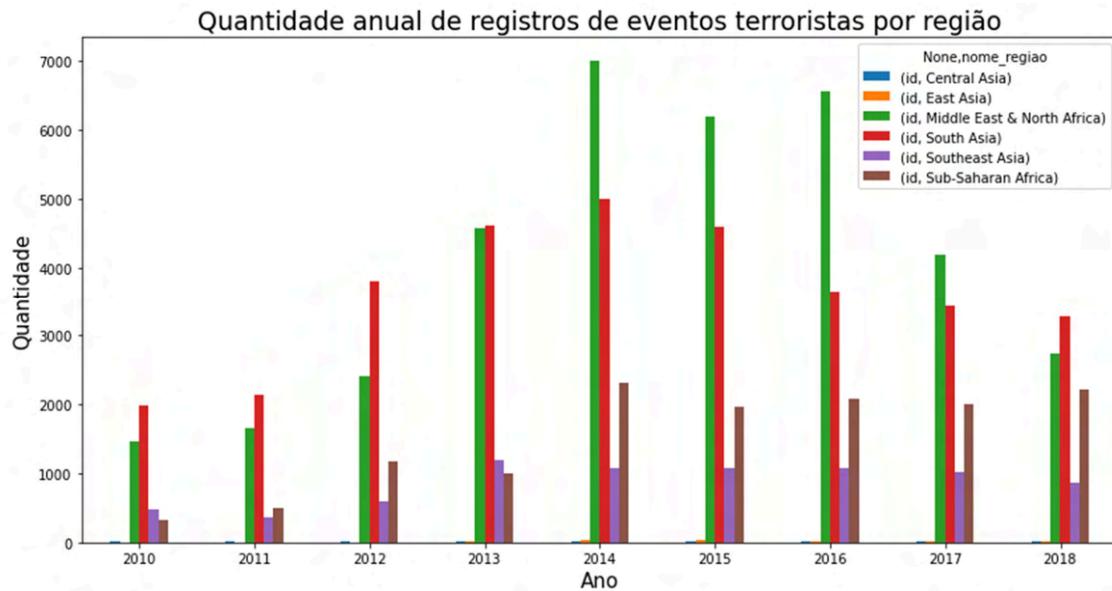
O alto número de registros na região do Oriente Médio e Norte da África se dá pela grande quantidade de eventos registrados no Iraque, o território que mais aparece na relação. Países como a Síria e o Iêmen, que enfrentaram guerras civis durante o período de estudo também estão localizados na região.



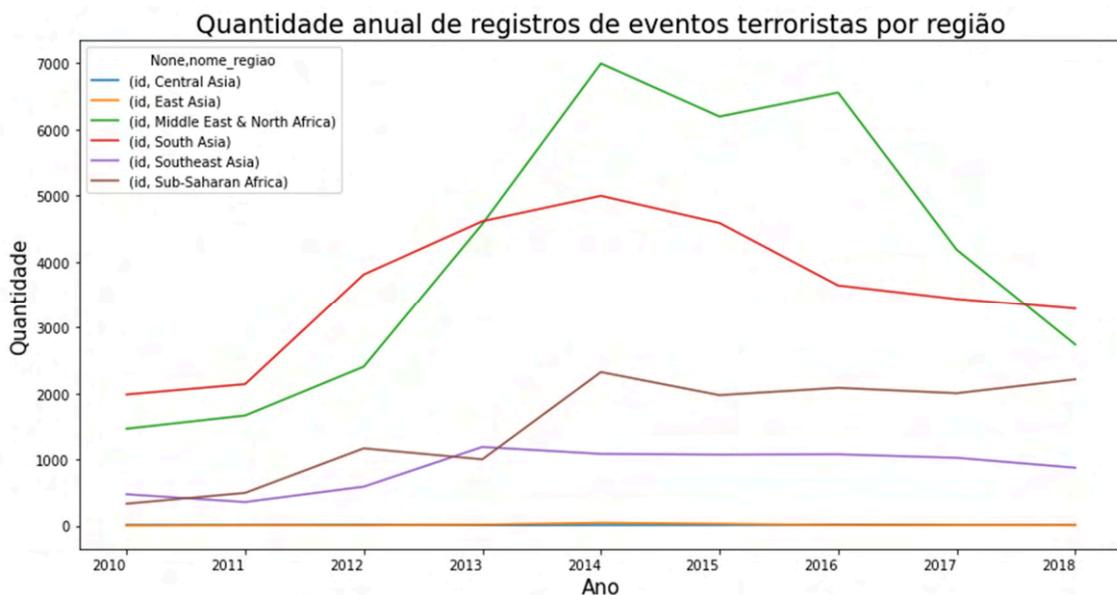
Observando as outras regiões que possuem índices elevados, como o sul da Ásia e a África subsaariana é possível relacionar os registros a eventos de guerras e conflitos do século passado ao início do século XXI.

Já na região leste da Ásia é possível identificar poucos registros, possibilitando assim, relacionar a quantidade de eventos terroristas com indicadores políticos elevados e índices sociais, como o índice de fracionalização étnica muito baixo em todos os países da região.

Na análise feita sobre as regiões estudadas, é possível perceber que, embora o auge do número de registros entre os continentes tenha sido no ano de 2014, nas regiões há uma variação local entre os anos do período estudado. As regiões do Oriente Médio e do Norte da África, do sul da Ásia, da África subsaariana e do leste asiático tiveram seu auge nesse período, enquanto a região central e a região sudeste da Ásia tiveram seu pico de registros em 2016 e 2013, respectivamente.



Outra importante observação a ser feita é que até o ano de 2013 o maior número de registros estava centralizado na região sul da Ásia, porém entre os anos de 2014 e 2017, a região com o maior número de registros foi a região do Oriente Médio e Norte da África, quando, novamente no ano de 2018, houveram maiores quantidades de registros de eventos terroristas na região sul da Ásia.



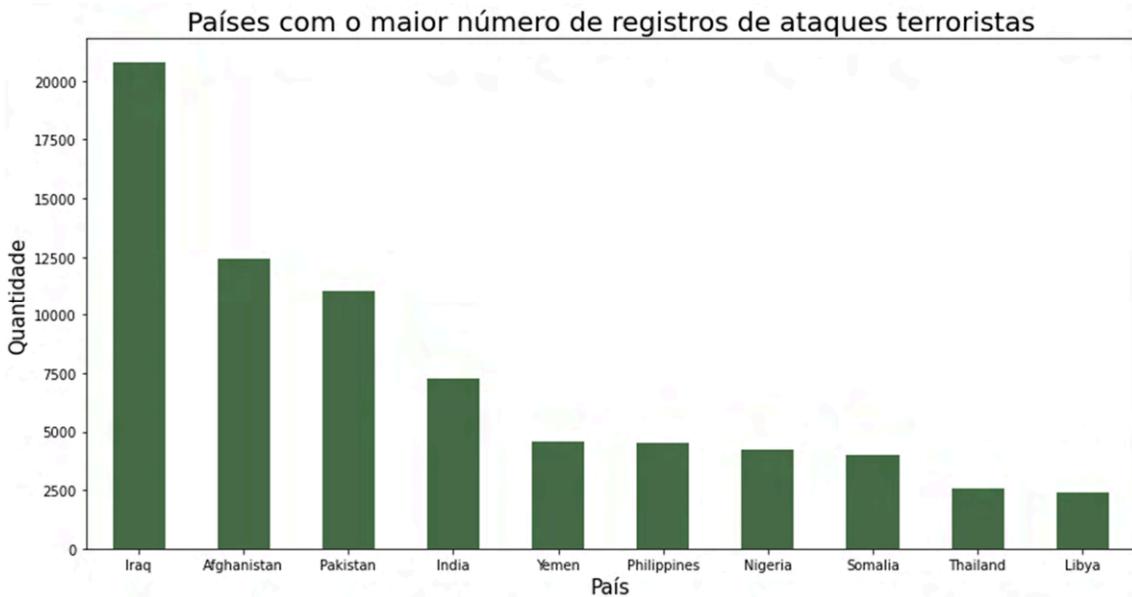
Aprofundando a pesquisa entre os territórios, pode-se perceber uma maior variação entre os territórios onde há números elevados de registros de ataques terroristas e as regiões estudadas. Dos dez territórios com os maiores números de registros, três estão localizados na região do Oriente Médio e Norte da África, três na região sul da Ásia, dois no sudeste asiático e dois na região da África subsaariana.

```
num_paises_regiao = analise_terrorismo.groupby(["nome_pais", "nome_regiao"])["id"].count().nlargest(10)
print("Países com o maior número de registros de ataques terroristas: " + str(num_paises_regiao))
```

| | Países com o maior número de registros de ataques terroristas: nome_pais | nome_regiao | |
|-------------|--|-------------|--|
| Iraq | Middle East & North Africa | 20766 | |
| Afghanistan | South Asia | 12434 | |
| Pakistan | South Asia | 11049 | |
| India | South Asia | 7241 | |
| Yemen | Middle East & North Africa | 4552 | |
| Philippines | Southeast Asia | 4501 | |
| Nigeria | Sub-Saharan Africa | 4225 | |
| Somalia | Sub-Saharan Africa | 4022 | |
| Thailand | Southeast Asia | 2578 | |
| Libya | Middle East & North Africa | 2402 | |

Name: id, dtype: int64

O Iraque, território localizado na região do Oriente Médio e Norte da África possui o maior número de registros de eventos terroristas com 20.766 registros, representando quase 23% dos registros totais. Seguido pelo Iraque, três territórios da região sul da Ásia apresentam números elevados de registros, são eles: Afeganistão, Paquistão e Índia, que juntos somam mais de 30.000 registros. Com números aproximados de registros, o Iêmen, país que atravessa uma guerra civil, é seguido pelas Filipinas, que contabilizam 4.552 e 4.501 registros respectivamente. Ao fim da lista dos 10 territórios com o maior número de registros, além da Tailândia, estão três países africanos: Nigéria e Somália na região da África subsaariana e a Líbia, país localizado na região norte da África.



É importante destacar que entre os dez territórios com o maior número de registros, grande parte deles sofre com guerras civis atuais ou sofreram com conflitos durante esse século. Pode-se destacar a Somália, onde uma guerra civil intensa se estende desde o século passado até atualmente. No continente asiático a invasão estadunidense ao Iraque no início do século levou a intensos conflitos, e mesmo após o fim da guerra em 2011, com a criação do Estado Islâmico no país, houve uma intensa disseminação da violência em territórios próximos, como a Síria.

Ainda sobre a análise voltada para os territórios, podemos mencionar as dez cidades com o maior número de registros. Dentre elas, três estão localizadas no Iraque. As cidades de Bagdá e Mosul, duas das três maiores cidades do país, encabeçam a lista, enquanto Quircuque (Kirkuk) aparece mais abaixo. Do Paquistão, três cidades também aparecem na relação: Karachi, na terceira posição, além de Quetta e Kabul. Apenas Arish, no Egito, pertence a um país que não figura entre os territórios com o maior número de registros.

```
num_cidade = analise_terrorismo.groupby([analise_terrorismo['cidade'][analise_terrorismo['cidade'] != 'Unknown']])['id'].count()
print("Cidades com o maior número de registros de ataques terroristas: " + str(num_cidade))

Cidades com o maior número de registros de ataques terroristas: cidade
Baghdad      5487
Mosul        1837
Karachi     1413
Mogadishu    1394
Benghazi     852
Quetta       678
Kirkuk       656
Kabul         611
Peshawar     606
Arish         540
Name: id, dtype: int64
```

Ao serem analisados quanto aos critérios utilizados para definir ou não se os eventos são considerados ataques terroristas, foi realizada a contabilização de registros para cada um dos critérios adotados pelo GTD. Para registros que não satisfazem o critério mencionado o valor é “0”, já para registros que satisfazem o critério mencionado o valor de identificação é “1”.

Os registros de eventos terroristas que satisfazem o primeiro critério somaram 89.940 ocorrências. Isto é, de 90.755 registros, pouco mais de 99% deles encaixam-se em motivações políticas, econômicas, religiosas ou sociais.

```
num_criterio1 = analise_terrorismo.groupby([analise_terrorismo['criterio1']])['id'].count()
print("Quantidade de registros de eventos terroristas que satisfazem ou não o primeiro critério: " + str(num_criterio1))

Quantidade de registros de eventos terroristas que satisfazem ou não o primeiro critério: criterio1
0      815
1    89940
Name: id, dtype: int64
```

Quanto ao segundo critério para classificação de um ato como um ataque terrorista foram contabilizados 90.271 registros. Dessa forma, aproximadamente 99,5% dos registros encaixam-se em atos com intenção de coagir, intimidar ou transmitir alguma mensagem para um público maior do que as vítimas imediatas.

```
num_criterio2 = analise_terrorismo.groupby([analise_terrorismo['criterio2']])['id'].count()
print("Quantidade de registros de eventos terroristas que satisfazem ou não o segundo critério: " + str(num_criterio2))

Quantidade de registros de eventos terroristas que satisfazem ou não o segundo critério: criterio2
0      484
1    90271
Name: id, dtype: int64
```

O número total de registros que satisfazem o terceiro critério foi um pouco menor se comparado aos outros dois critérios, mas mesmo assim, 84,5% dos registros

são considerados atos realizados fora do contexto de guerra. O número de eventos classificados dessa maneira é 76.704 registros.

```
num_criterio3 = analise_terrorismo.groupby([analise_terrorismo['criterio3']])['id'].count()
print("Quantidade de registros de eventos terroristas que satisfazem ou não o terceiro critério: " + str(num_criterio3))

Quantidade de registros de eventos terroristas que satisfazem ou não o terceiro critério: criterio3
0    14051
1    76704
Name: id, dtype: int64
```

Após uma análise geral dos registros, pode-se observar que cerca de 83% dos registros de eventos terroristas, ou 75.405 dos registros, satisfazem os três critérios propostos pelo GTD para classificação de um ato como ataque terrorista.

```
#filtro = analise_terrorismo['nome_continente'] == 'Africa' and analise_terrorismo['nome_continente'] == 'Asia'
num_criterio_todos = analise_terrorismo[(analise_terrorismo['criterio1'] == 1) & (analise_terrorismo['criterio2'] == 1) & (analise_terrorismo['criterio3'] == 1)]
print("Quantidade de registros de eventos terroristas que satisfazem todos os critérios: " + str(num_criterio_todos))

Quantidade de registros de eventos terroristas que satisfazem todos os critérios: 75405
```

Porém, nem todos os eventos são precisamente caracterizados e classificados como ataque terrorista. De acordo com base de dados desenvolvida pelo GTD há registros onde há grande possibilidade de serem classificados como ataques terroristas, porém não há certeza sobre essa categorização.

```
num_duvida = analise_terrorismo.groupby([analise_terrorismo['duvida_terrorismo']])['id'].count()
print("Quantidade de registros de eventos terroristas quanto a convicção sobre sua classificação \n0 -> Não há dúvidas que foi um ataque terrorista\n1 -> Há dúvidas se realmente foi um ataque terrorista:duvida_terrorismo")

Quantidade de registros de eventos terroristas quanto a convicção sobre sua classificação
0 -> Não há dúvidas que foi um ataque terrorista
1 -> Há dúvidas se realmente foi um ataque terrorista:duvida_terrorismo
0    74808
1    15947
Name: id, dtype: int64
```

Durante o período pesquisado pouco mais de 17,5% dos registros foram classificados como ataques terroristas, mas que ainda apresentavam dúvidas sobre a sua correta classificação.

Há casos em que diversos ataques estão conectados, porém não necessariamente constituem um único incidente, geralmente onde o momento da ocorrência ou a sua localização são descontínuas.

```
num_attaques_conectados = analise_terrorismo.groupby([analise_terrorismo['ataques_conectados']])['id'].count()
print("Quantidade de registros de eventos terroristas que estão relacionados ou não a outros eventos \n0 -> Não há relação com outros eventos\n1 -> Há relação com outros eventos: ataques_conectados")

Quantidade de registros de eventos terroristas que estão relacionados ou não a outros eventos
0 -> Não há relação com outros eventos
1 -> Há relação com outros eventos: ataques_conectados
0    74842
1    15913
Name: id, dtype: int64
```

No período analisado na pesquisa apenas pouco mais de 17,5% dos ataques estão conectados a outros incidentes. A maioria dos eventos registrados, cerca de 82,5% dos registros, são considerados apenas como um incidente.

O êxito de um ataque terrorista é definido de acordo como os efeitos tangíveis do ataque. Nesse caso, o êxito não é definido necessariamente pelo objetivo absoluto

do grupo perpetrador. A “taxa de êxito” depende essencialmente do tipo de ataque realizado. Por exemplo, execuções só são bem-sucedidas se o alvo pretendido for de fato morto, porém um bombardeio a um prédio pode ser considerado bem-sucedido mesmo que o prédio não seja derrubado.

```
num_sucesso_ataque = analise_terrorismo.groupby(['sucesso_ataque'])['id'].count()
print("Quantidade de registros de eventos terroristas onde houve sucesso no ataque\n0 -> Não houve sucesso no ataque\n1 -> Houve sucesso no ataque: sucesso_ataque")
0    13019
1    77736
Name: id, dtype: int64
```

Durante o período analisado mais de 85% dos ataques foram considerados bem-sucedidos. Mais à frente será possível observar a taxa de êxito para cada método de ataque.

Ao realizar uma análise mais profunda nos registros é possível obter informações importantes, como os métodos utilizados durante os ataques, principais perfis de alvos e vítimas, e também de grupos perpetradores e o número de óbitos e feridos decorrentes de eventos terroristas.

De acordo com o GTD os métodos utilizados durante os ataques são classificados em oito categorias (além da categoria desconhecido):

- Execução
- Ataques armados a população
- Bombardeios e explosões
- Sequestro de veículos
- Sequestro de pessoas mantidas como reféns
- Sequestro de pessoas levadas a cativeiro em outro local
- Ataques a instalações, como prédios e monumentos
- Ataques não armados, mas que utilizam outros meios de atingir a população, como armas químicas, radiológicas e biológicas
- Desconhecido. O método de ataque não foi claramente identificado.

Durante a análise sobre os métodos utilizados em eventos terroristas pode-se destacar bombardeio e explosões como sendo o principal recurso de grupos perpetradores. O método mencionado está presente em 46.303 ocorrências, mais da metade dos registros. A possibilidade de se atingir um nível maior de estrago, além de surpreender as vítimas através desse tipo de ataque pode indicar a opção por esse recurso.

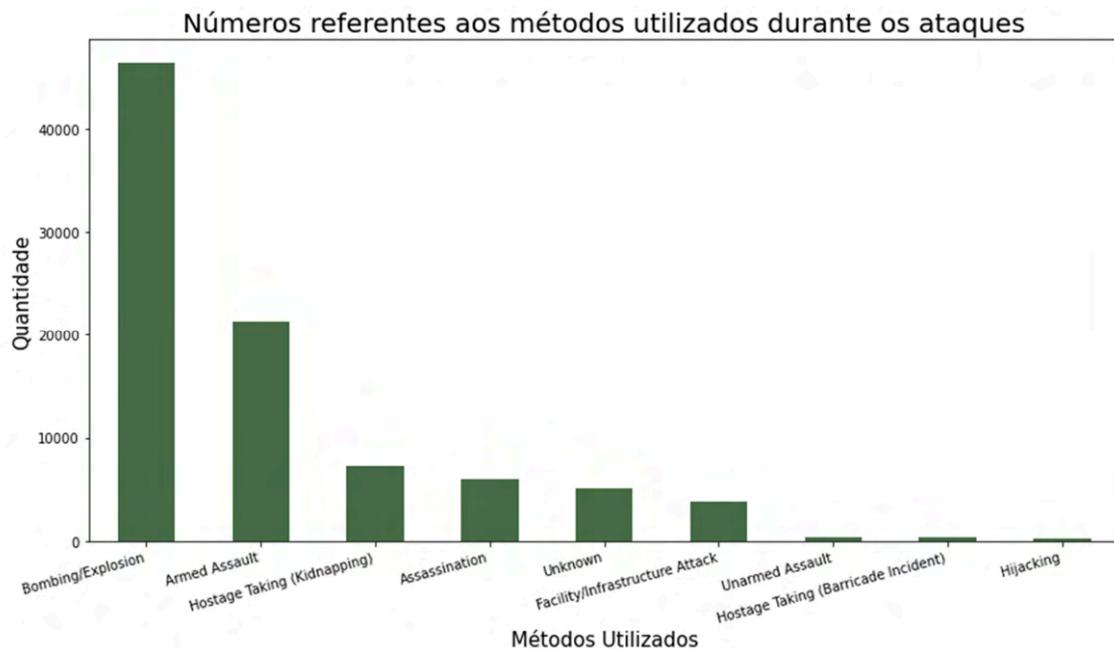
```

num_metodo_ataque = analise_terrorismo.groupby(['metodo_ataque'])['id'].count().sort_values(ascending=False)
print("Números referentes aos métodos utilizados durante os ataques: " + str(num_metodo_ataque))

Números referentes aos métodos utilizados durante os ataques: metodo_ataque
Bombing/Explosion      46303
Armed Assault           21276
Hostage Taking (Kidnapping) 7223
Assassination            6046
Unknown                  5114
Facility/Infrastructure Attack 3815
Unarmed Assault          393
Hostage Taking (Barricade Incident) 345
Hijacking                 240
Name: id, dtype: int64

```

Com mais de 21.000 ocorrências, os ataques armados à população, utilizando em sua maioria explosivos e armas de fogo, é o item seguinte na lista de métodos utilizados durante ataques terroristas. Em seguida, o sequestro de pessoas levadas a cativeiro e execuções são os itens que aparecem em grande parte das ocorrências, em 7.223 e 6.046 registros, respectivamente.



Embora o método de bombardeio e explosões esteja presente na maioria dos registros, é o método de sequestro de pessoas mantidas reféns que apresenta a maior taxa de êxito entre os ataques, seguidos por sequestro de pessoas levadas a cativeiro em outro local e ataques a instalações.

```

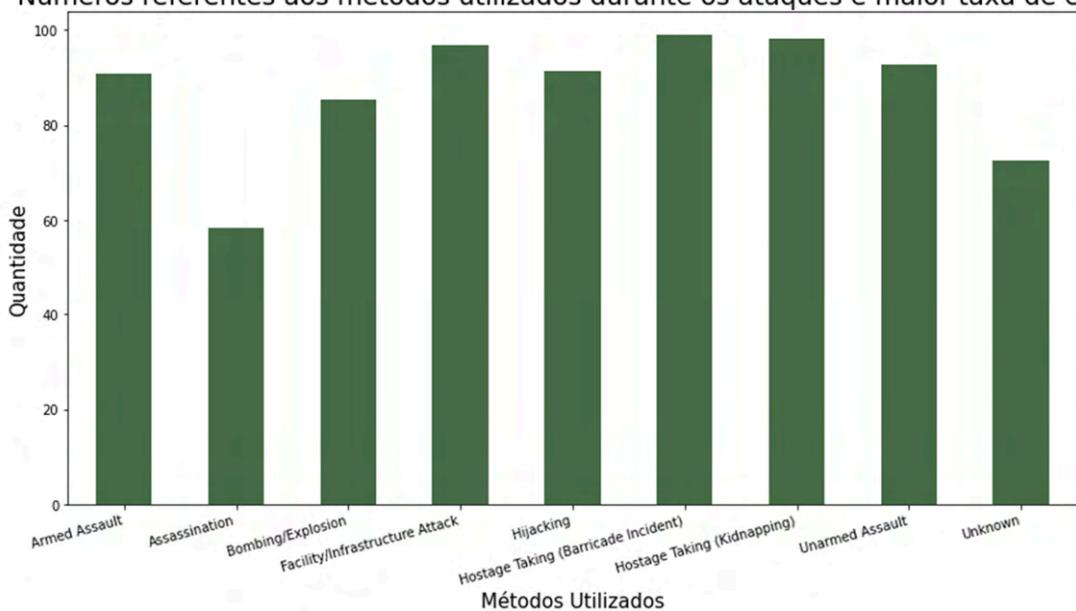
perc_metodo_sucesso = num_metodo_sucesso *100 / num_metodo_ataque
print("Porção de êxito de ataque por quantidade de registros:" + str(perc_metodo_sucesso))

Porcentagem de êxito de ataque por quantidade de registros:metodo_ataque
Armed Assault           90.844144
Assassination            58.451869
Bombing/Explosion        85.232058
Facility/Infrastructure Attack 96.671035
Hijacking                 91.250000
Hostage Taking (Barricade Incident) 98.840580
Hostage Taking (Kidnapping) 98.214038
Unarmed Assault          92.620865
Unknown                   72.409073
Name: id, dtype: float64

```

Através do gráfico é possível observar a proporção de êxito de cada método utilizado nos ataques terroristas.

Números referentes aos métodos utilizados durante os ataques e maior taxa de êxito



Apenas o método de execução apresenta um nível abaixo dos 70% de êxito entre todos os métodos. O método de bombardeios e explosões tem uma taxa de êxito de pouco mais de 85% e aparece apenas como o antepenúltimo entre todos os métodos, embora seja o que mais aparece na relação.

Os alvos ou vítimas de ataques terroristas estão classificados em vinte e uma categorias de acordo com o GTD. De maneira geral, os principais alvos são cidadãos e propriedades particulares, porém analisando de forma mais profunda os registros pode-se perceber que os principais alvos de grupos perpetradores estão ligados a forças militares, governamentais e policiais dos territórios.

```
num_tipo_vitima_compc = analise_terrorismo.groupby(['tipo_vitima'])[['id']].count().nlargest(12).sort_values(as
print("Quantidade de registros de eventos terroristas por tipo de vítima/alvo: " + str(num_tipo_vitima_compc))
|
```

| Tipo de Vítima | Quantidade |
|--------------------------------|------------|
| Private Citizens & Property | 24633 |
| Military | 17405 |
| Police | 13903 |
| Government (General) | 9075 |
| Business | 6380 |
| Unknown | 5076 |
| Educational Institution | 2284 |
| Religious Figures/Institutions | 2152 |
| Terrorists/Non-State Militia | 2142 |
| Transportation | 1829 |
| Utilities | 1585 |
| Violent Political Party | 1087 |
| Name: id, dtype: int64 | |

Analizando todas as classificações de vítimas e alvos através do gráfico é possível observar que alguns serviços essenciais para população não são considerados alvos muito frequentes em ataques terroristas.



Os métodos de comunicação como rádio e televisão não são alvos principais em eventos desse tipo. O mesmo ocorre para serviços de alimentação e abastecimento de água e portos e o comércio marítimo. Diferentemente do ataque a meios de transporte, como metrôs, trens e ônibus que figuram mais acima na relação.

Quando se trata da especificação e identificação dos alvos mais atingidos, instituições de defesa da população e interesses governamentais encabeçam a lista. Analisando o perfil dos doze tipos de alvos mais recorrentes, é possível ver que a metade deles está ligada a forças de proteção civil e territorial, como postos policiais, bases militares e pessoas ligadas a forças de proteção. Enquanto isso, pessoas ligadas ao governo e partidos políticos (tanto um indivíduo, quanto um grupo) aparecem duas vezes nessa lista.

```
num_subtipo_vitima = analise_terrorismo.groupby(['subtipo_vitima'])[['id']].count().nlargest(12)
print("Quantidade e especificação das vítimas/alvos que aparecem com mais frequência nos registros " + str(num_subtipo_vitima))

Quantidade e especificação das vítimas/alvos que aparecem com mais frequência nos registros subtipo_vitima
Unnamed: 0
Unnamed Civilian/Unspecified          7400
Unknown                                5845
Military Personnel (soldiers, troops, officers, forces) 5630
Police Security Forces/Officers        5582
Village/City/Town/Suburb              4369
Military Barracks/Base/Headquarters/Checkpost 3784
Government Personnel (excluding police, military) 3274
Police Patrol (including vehicles and convoys)    3269
Police Building (headquarters, station, school)   3175
Military Unit/Patrol/Convoy            3142
Politician or Political Party Movement/Meeting/Rally 2526
Marketplace/Plaza/Square                1834
Name: id, dtype: int64
```

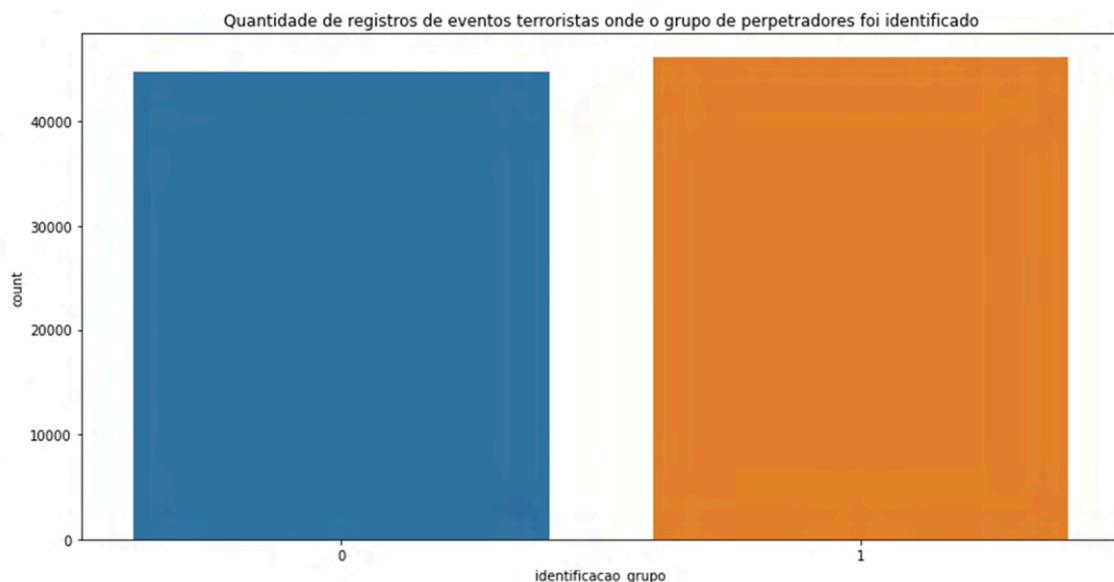
Além disso, há de se destacar lugares públicos como mercados, shoppings e praças. Ademais, ataques a locais públicos podem ser mais centralizados em cidades grandes, com o objetivo de causar maior impacto; e em vilas e cidades menores,

geralmente com motivações étnicas e sociais, como perseguições a pequenos grupos religiosos e de diferentes ideologias.

Durante a investigação dos registros podemos observar que em pouco mais da metade deles o grupo de perpetradores foi identificado, seja por meio da identificação através da inteligência governamental ou por uma própria reivindicação do grupo.

```
identificacao_grupo = analise_terrorismo.groupby([analise_terrorismo['identificacao_grupo']])['id'].count()
print("Quantidade de registros onde o grupo de perpetradores foi identificado\n0 -> Não foram identificados\n1 -> Foram identificados")
print(identificacao_grupo)
print("\nName: id, dtype: int64")
```

Quantidade de registros onde o grupo de perpetradores foi identificado
0 -> Não foram identificados
1 -> Foram identificados: identificacao_grupo
0 44682
1 46073
Name: id, dtype: int64



Dentre os grupos identificados, o Talibã, grupo que rege o atual governo do Afeganistão e que foi ligado ao ataque do 11 de Setembro, tendo como resposta a chamada “Guerra ao Terrorismo” do governo estadunidense, está no topo da lista, sendo mencionado em 7.590 registros. O segundo grupo com mais menções é o atual Estado Islâmico (ISIS), grupo que atua no Oriente Médio, em especial, no Iraque e na Síria com 6.387 registros. A seguir na lista, os dois grupos mais atuantes na África subsaariana, o grupo radical Al-Shabaab na Somália e o Boko Haram, grupo militante islâmico conhecido pela atuação em países como a Nigéria, Níger e Camarões.

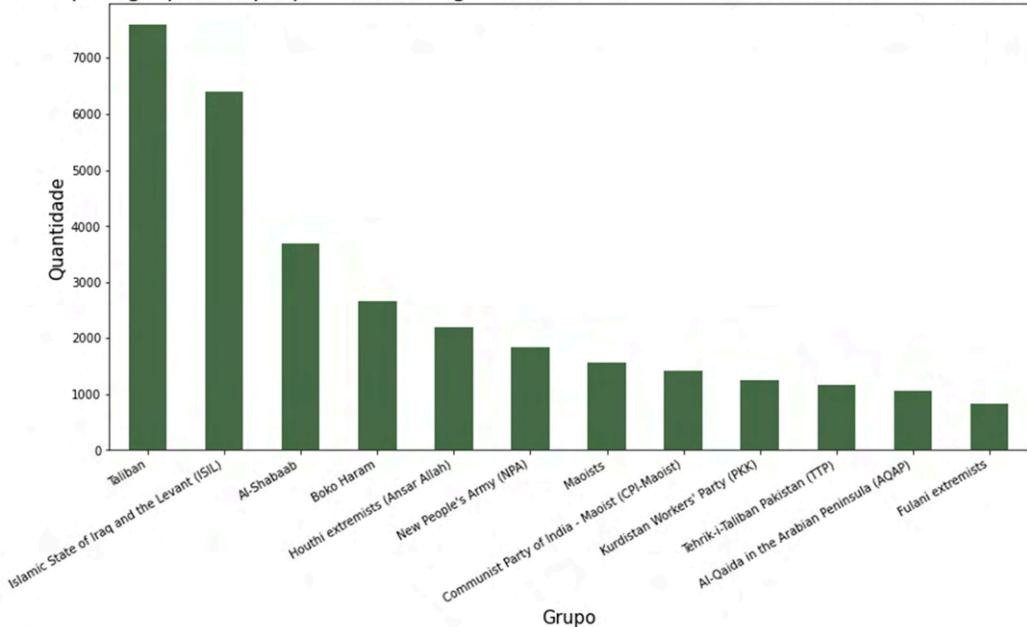
```

num_grupo_terrorista_mundo = analise_terrorismo.groupby(['analise_terrorismo['grupo_terrorista']'][analise_terrorismo['grupo_terrorista'].str != 'Unknown']].sum()
print("Principais grupos de perpetradores registrados em eventos terroristas (0 valor 'Unknown' foi desconsiderado): " + str(num_grupo_terrorista_mundo))
print('')

Principais grupos de perpetradores registrados em eventos terroristas (0 valor 'Unknown' foi desconsiderado): grupo_terrorista
Taliban                                7590
Islamic State of Iraq and the Levant (ISIL)    6387
Al-Shabaab                               3686
Boko Haram                               2663
Houthi extremists (Ansar Allah)           2193
New People's Army (NPA)                  1824
Maoists                                  1554
Communist Party of India - Maoist (CPI-Maoist) 1415
Kurdistan Workers' Party (PKK)            1236
Tehrik-i-Taliban Pakistan (TTP)          1156
Al-Qaida in the Arabian Peninsula (AQAP) 1063
Fulani extremists                         819
Name: id, dtype: int64

```

Principais grupos de perpetradores registrados em eventos terroristas (sem o valor 'Unknown')



Analisando os registros e contabilizando o número de óbitos podemos relacionar os dados a quantidade de registros de eventos terroristas. Assim como houve um maior número de registros de eventos terroristas em 2014, o ano também registra o maior número total de mortos, com 42.850 óbitos. Esse número é maior que os registros entre 2010 e 2013 somados. Em seguida, o ano de 2015 com 37.838 e 2016 com 34.785 aparecem na relação. O ano de 2010 registrou o menor número de óbitos, 7.527 no total.

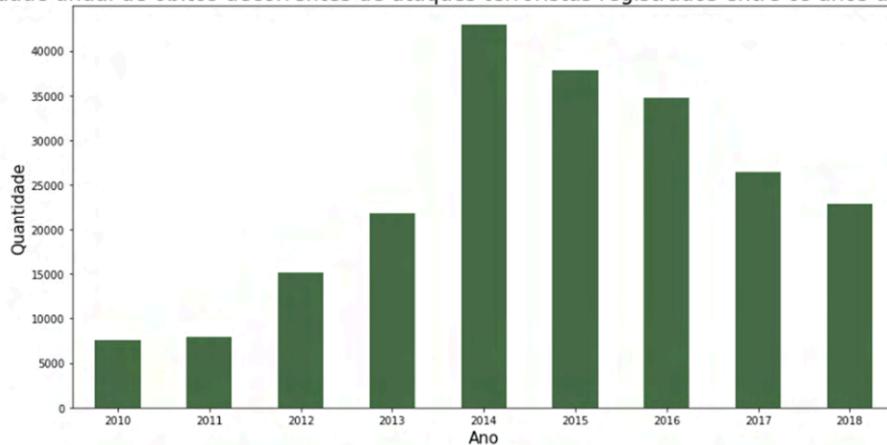
```

num_obitos_anuais = analise_terrorismo.groupby(['ano'])['numero_obitos'].sum()
print("Quantidade anual de óbitos decorrentes de ataques terroristas registrados entre os anos de 2010 e 2018: " + str(num_obitos_anuais))
print('')

Quantidade anual de óbitos decorrentes de ataques terroristas registrados entre os anos de 2010 e 2018: ano
2010      7527
2011     7944
2012    15095
2013    21883
2014    42850
2015    37838
2016    34785
2017    26472
2018    22930
Name: numero_obitos, dtype: int64

```

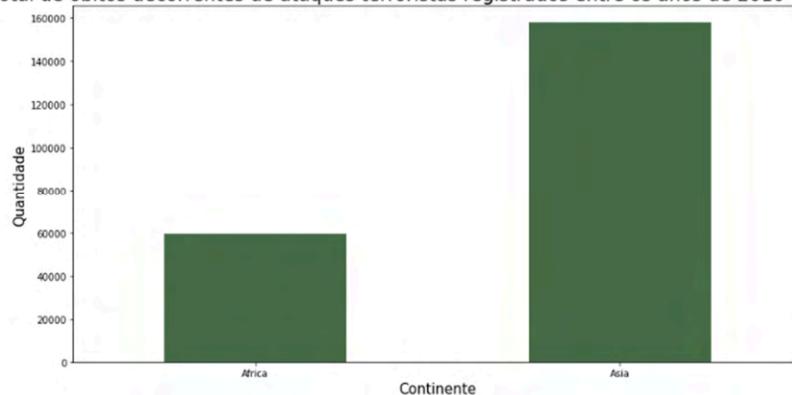
Quantidade anual de óbitos decorrentes de ataques terroristas registrados entre os anos de 2010 e 2018



Em relação aos continentes, o número de óbitos na Ásia é mais de duas vezes maior que na África.

```
num_obitos_continente = analise_terrorismo.groupby(['nome_continente'])['numero_obitos'].sum()
print("Quantidade total de óbitos decorrentes de ataques terroristas registrados entre os anos de 2010 e 2018 nos continentes pesquisados: nome_continente")
Africa      59605
Asia       157719
Name: numero_obitos, dtype: int64
```

Quantidade total de óbitos decorrentes de ataques terroristas registrados entre os anos de 2010 e 2018 por continente



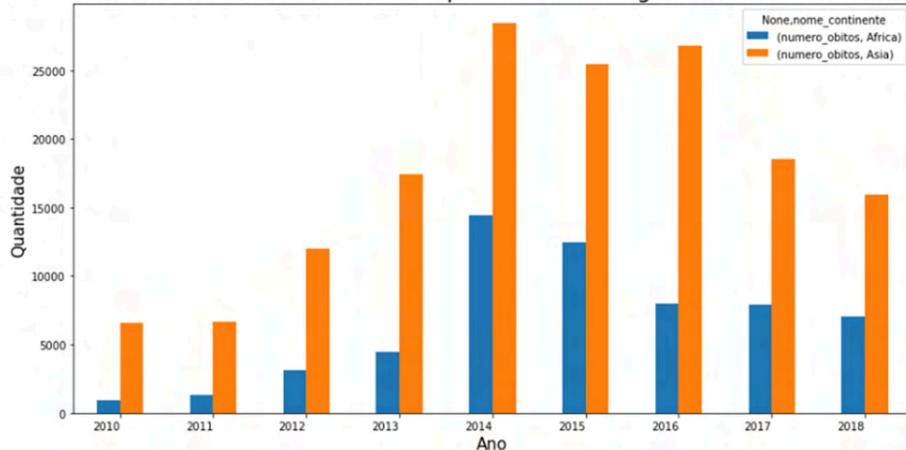
Na análise entre os continentes durante o período estudado é possível observar que ambos tiveram o seu pico de registros de óbitos no ano de 2014, enquanto no ano de 2010 registraram o menor número no período.

```

num_obitos_continente_anuais = analise_terrorismo.groupby([analise_terrorismo['ano'],analise_terrorismo['nome_continente']])['numero_obitos'].sum()
print("Quantidade anual de óbitos decorrentes de ataques terroristas registrados entre os anos de 2010 e 2018 nos continentes pesquisados: ano nome_continente")
2010 Africa 939
      Asia 6588
2011 Africa 1311
      Asia 6633
2012 Africa 3105
      Asia 11990
2013 Africa 4460
      Asia 17423
2014 Africa 14468
      Asia 28382
2015 Africa 12411
      Asia 25427
2016 Africa 7984
      Asia 26881
2017 Africa 7928
      Asia 18544
2018 Africa 6999
      Asia 15931
Name: numero_obitos, dtype: int64

```

Quantidade anual de óbitos decorrentes de ataques terroristas registrados nos continentes pesquisados



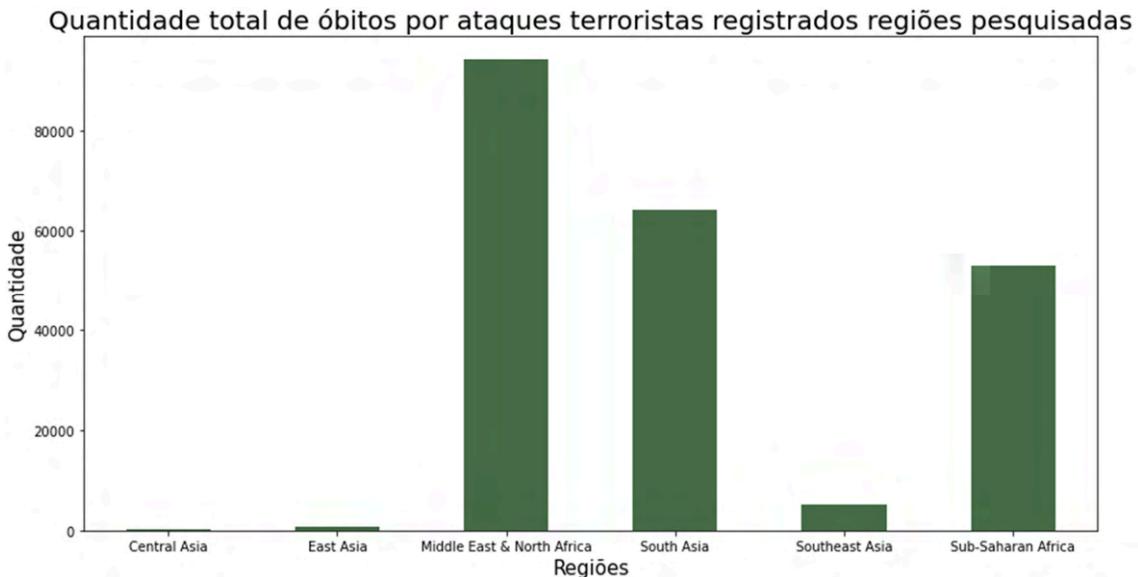
Ao analisar o gráfico comparativo entre os dois continentes em todo o período, é possível observar que o número de óbitos no continente africano cresceu bruscamente no ano de 2014, porém nos anos seguintes registrou quedas nos registros. Já no continente asiático houve uma variação após 2014. Após a queda de registros em 2015, o número de óbitos no ano seguinte voltou a ser maior que no ano anterior, até voltar a diminuir entre 2017 e 2018.

O número de óbitos entre as regiões segue o número de registros, sendo a região do Oriente Médio e Norte da África com o maior número de mortes, com 94.184 registros, com uma média superior a 10.000 mortes anuais no período da pesquisa. Em seguida, a região sul da Ásia apresenta o segundo maior número de óbitos com 64.142 registros, seguido pela região da África Subsaariana, com 53.061 registros.

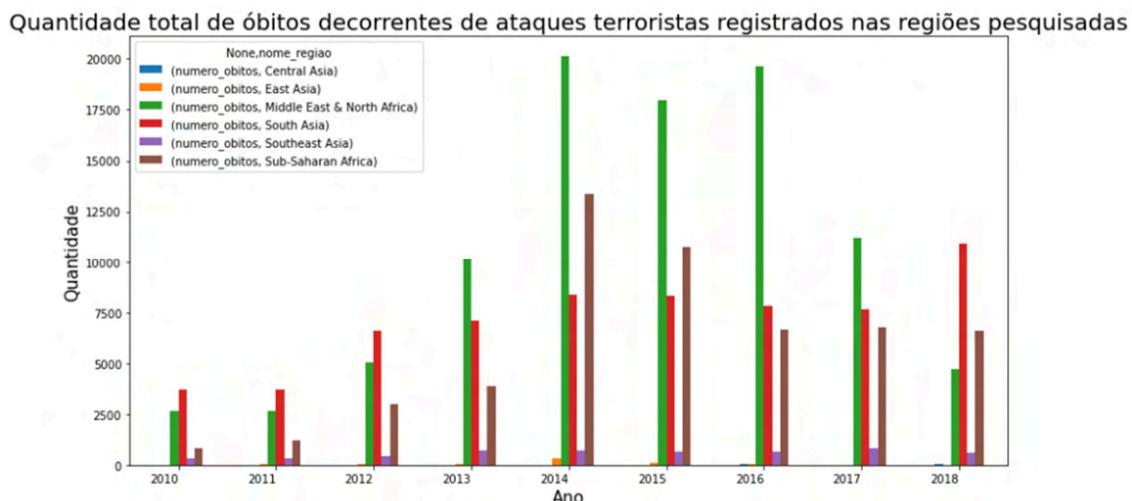
```

num_obitos_regioes = analise_terrorismo.groupby(['nome_Regiao'])['numero_obitos'].sum()
print("Quantidade total de óbitos decorrentes de ataques terroristas registrados entre os anos de 2010 e 2018 nas regiões pesquisadas:")
print(nome_Regiao)
Central Asia           109
East Asia              688
Middle East & North Africa 94184
South Asia             64142
Southeast Asia          5220
Sub-Saharan Africa      53061
Name: numero_obitos, dtype: int64

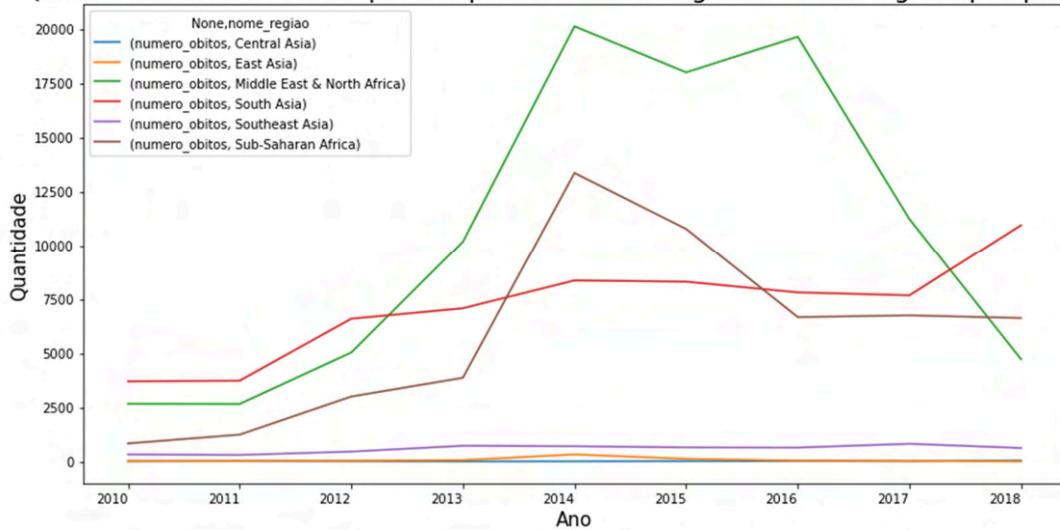
```



Investigando os registros de óbitos por ano, percebe-se uma variação entre o pico de registros de óbitos em cada região. As regiões do Oriente Médio e Norte da África, leste da Ásia e a região da África subsaariana registraram o maior número de óbitos no ano de 2014. Já as regiões central e sul da Ásia registraram a maior quantidade de óbitos no ano de 2018, enquanto a região do sudeste asiático registrou em 2017 o seu pico de registros de óbitos por ataques terroristas.

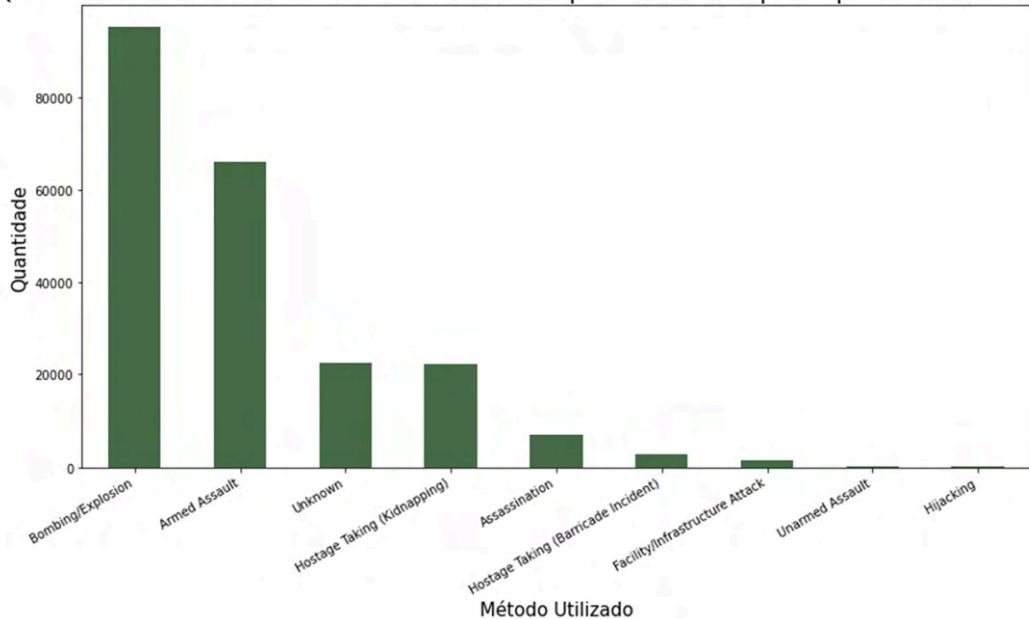


Quantidade total de óbitos por ataques terroristas registrados nas regiões pesquisadas



Os métodos de ataque mais recorrentes entre os registros acabam sendo os mais fatais. A opção por bombardeios e explosões matou no período estudado pouco mais de 95.000 pessoas. Em seguida, a utilização de armas de fogo e incendiárias estão ligadas a 65.900 mortes. O número de mortes decorrentes de ataques terroristas, mas por métodos desconhecidos totalizam 22.599 registros, pouco a mais que o número de pessoas mortas em sequestros e levadas a cativeiro, que registram pouco mais de 22.200 óbitos.

Quantidade total de óbitos decorrentes de ataques terroristas pelo tipo de método utilizado



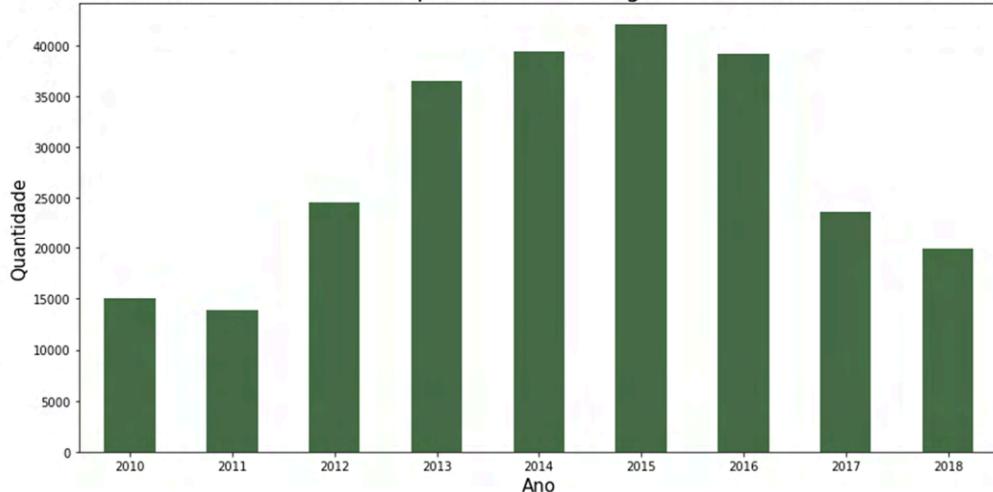
Durante a exploração dos dados de feridos entre o período pesquisado, podemos identificar uma variação nos números. Embora a maior quantidade de eventos terroristas e óbitos tenham sido registrados no ano de 2014, o ano de 2015 é o que registra o maior número de feridos, com 42.028 registros, seguido pelos anos

de 2014 e 2016, com leve diferença entre ambos, registrando cada um 39.363 e 39.214, respectivamente. Ao contrário dos registros de óbitos analisados, o menor número de feridos não foi contabilizado em 2010, mas sim em 2011, com 13.878 registros.

```
num_feridos_anuais = analise_terrorismo.groupby(['ano'])['numero_feridos'].sum()
print("Quantidade anual de feridos registrados entre os anos de 2010 e 2018: " + str(num_feridos_anuais))

Quantidade anual de feridos registrados entre os anos de 2010 e 2018: ano
2010    15106
2011    13878
2012    24688
2013    36468
2014    39363
2015    42028
2016    39214
2017    23588
2018    19932
Name: numero_feridos, dtype: int64
```

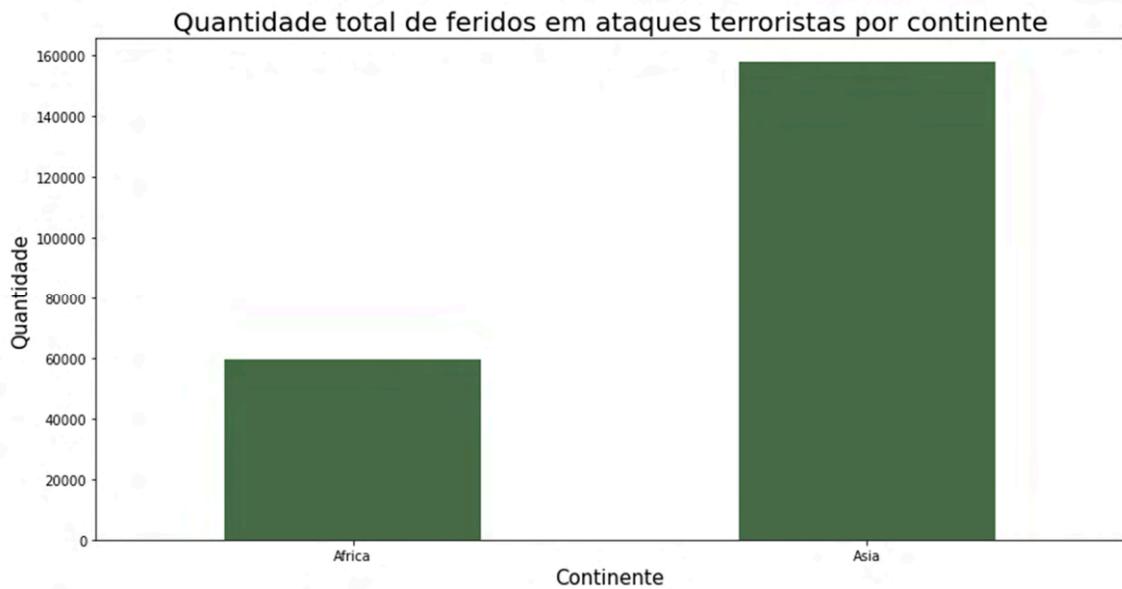
Quantidade anual de feridos em ataques terroristas registrados entre os anos de 2010 e 2018



Analizando os registros e a contabilização de feridos em ataques terroristas durante o período pesquisado, é possível compará-los também aos registros de óbitos nos continentes. Enquanto o número de feridos no continente asiático supera o número de óbitos, no continente africano o número de feridos é menor que o número de óbitos registrados. Em um panorama geral, o número de feridos na Ásia é mais de 5 vezes maior que o número de feridos na África.

```
num_feridos_regioes = analise_terrorismo.groupby(['nome_continente'])['numero_feridos'].sum()
print("Quantidade anual de feridos em eventos terroristas nos continentes africano e asiático: nome_continente")

Quantidade anual de feridos em eventos terroristas nos continentes africano e asiático: nome_continente
Africa      40775
Asia       213410
Name: numero_feridos, dtype: int64
```

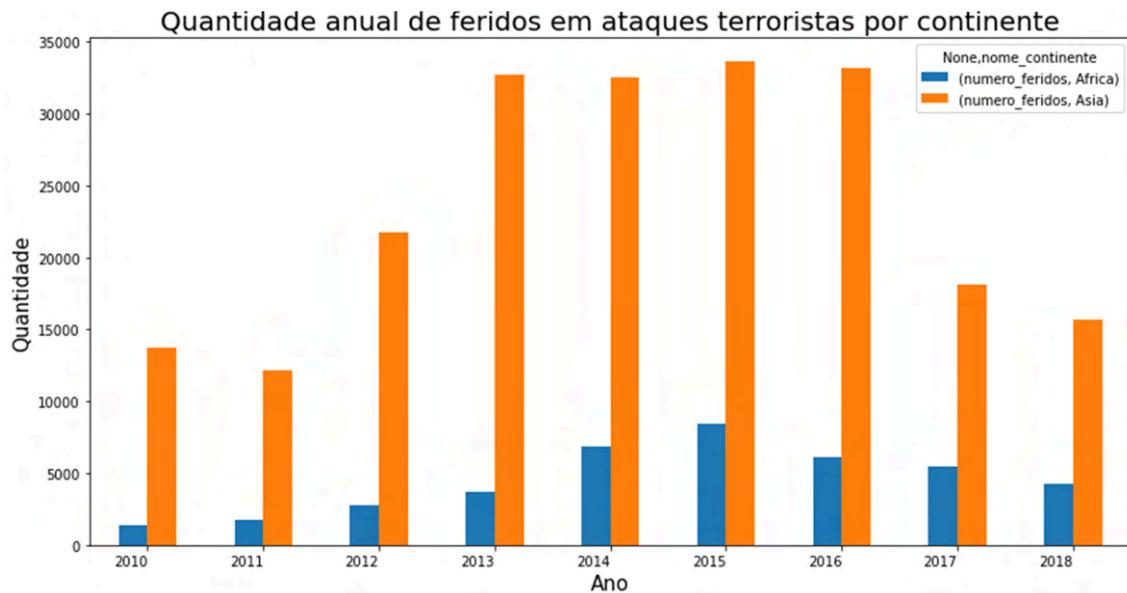


Tanto o continente asiático quanto o africano tiveram os maiores registros de feridos no ano de 2015. O continente asiático registrou 33.594 feridos enquanto o continente africano registrou 8.434 no mesmo ano. Em relação ao período onde houve o menor número de feridos, porém, não houve semelhança entre ambos. A menor quantidade de feridos na Ásia foi registrada no ano de 2011, totalizando 12.096 feridos, enquanto na África, o ano de 2010 é o período onde foram registrados o menor número de feridos, totalizando 1.362 registros.

```

num_obitos_Regioes_anuais = analise_terrorismo.groupby([analise_terrorismo['ano'],analise_terrorismo['nome_continente']])['numero_'
print("Quantidade anual de feridos em ataques terroristas por continente: " + str(num_obitos_Regioes_anuais))
<
Quantidade anual de feridos em ataques terroristas por continente: ano    nome_continente
2010  Africa      1362
      Asia       13744
2011  Africa      1782
      Asia       12096
2012  Africa      2807
      Asia       21801
2013  Africa      3724
      Asia       32744
2014  Africa      6855
      Asia       32508
2015  Africa      8434
      Asia       33594
2016  Africa      6057
      Asia       33157
2017  Africa      5465
      Asia       18123
2018  Africa      4289
      Asia       15643
Name: numero_feridos, dtype: int64

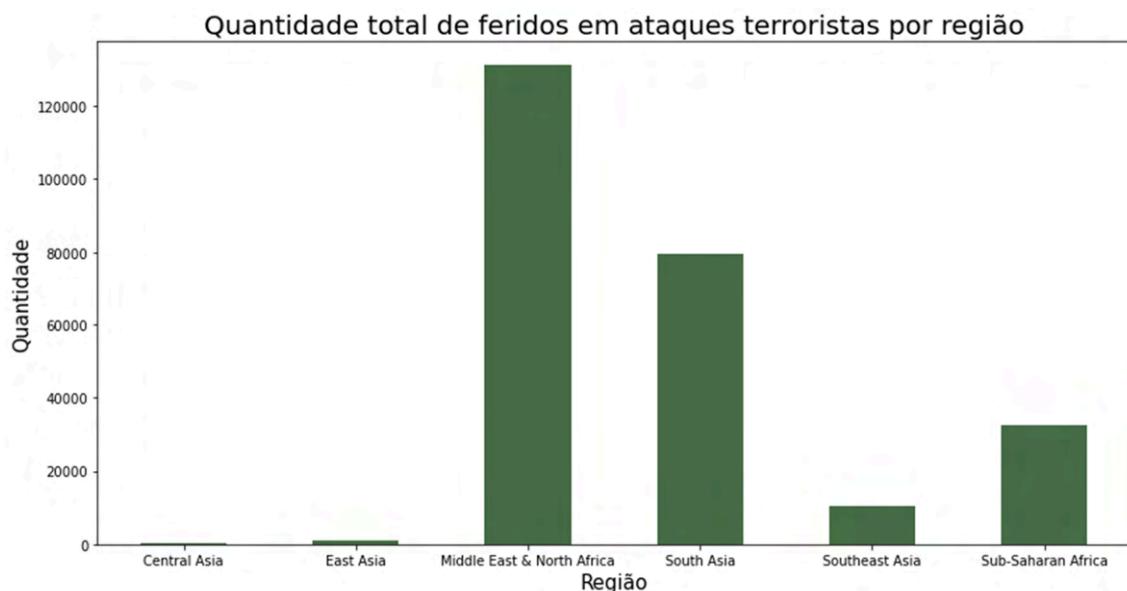
```



O número de feridos entre as regiões segue o número de registros de ataques terroristas e óbitos, sendo a região do Oriente Médio e Norte da África registrando maior número de feridos, com 131.083 registros. Em seguida, a região sul da Ásia apresenta o segundo maior número de feridos com 79.454 registros, seguido pela região da África Subsaariana, com 32.450 registros.

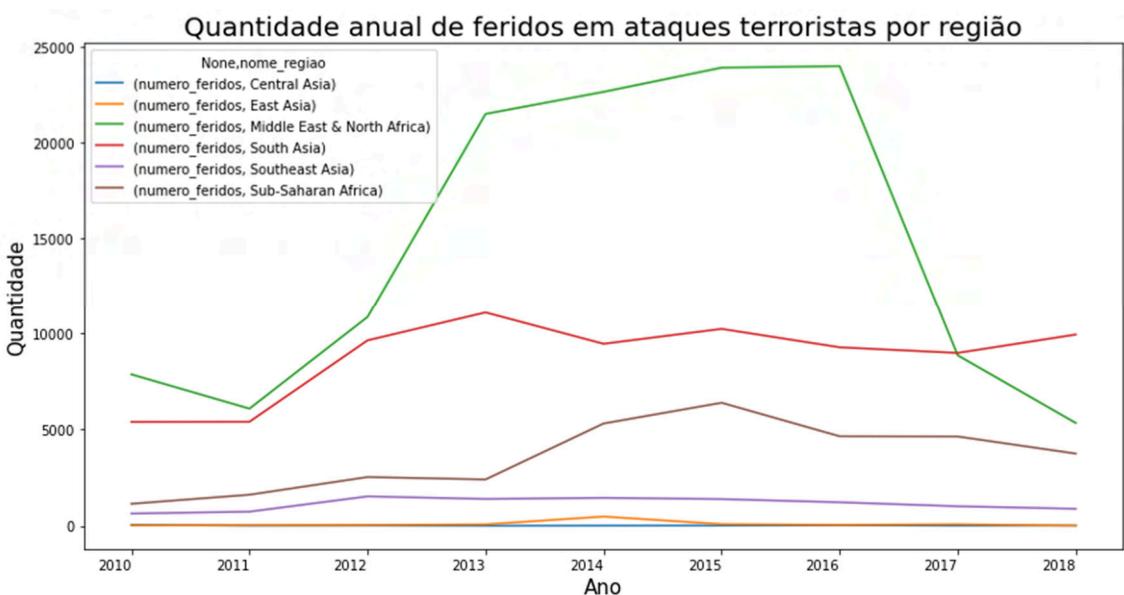
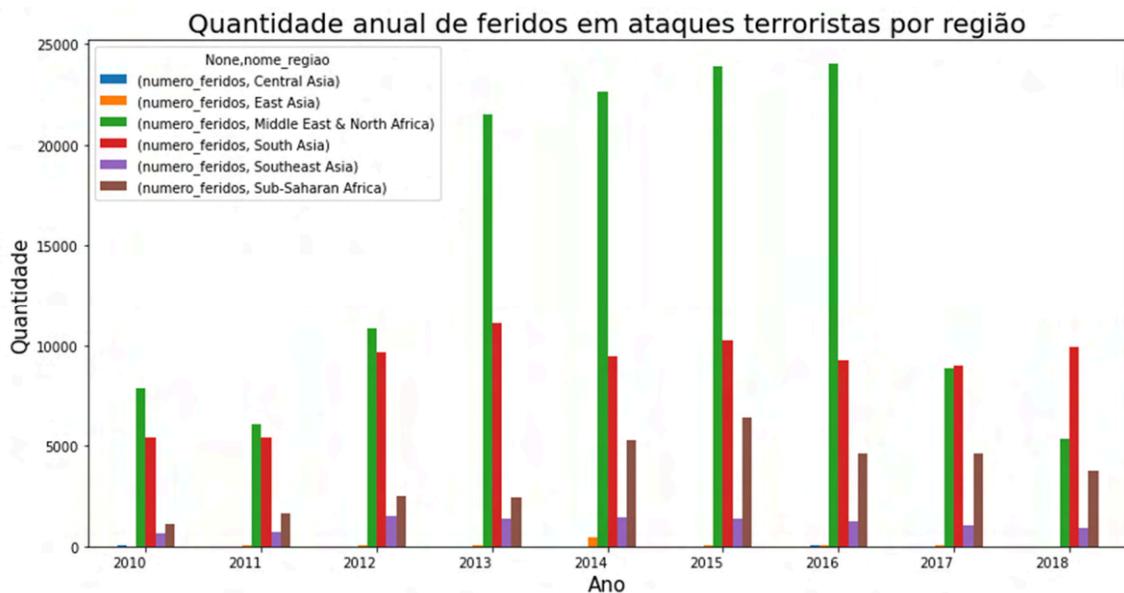
```
num_feridos_regioes = analise_terrorismo.groupby(['nome_Regiao'])['numero_feridos'].sum()
print("Quantidade total de feridos em ataques terroristas por região: " + str(num_feridos_regioes))
```

```
Quantidade total de feridos em ataques terroristas por região: nome_Regiao
Central Asia           120
East Asia              840
Middle East & North Africa 131883
South Asia             79454
Southeast Asia          10238
Sub-Saharan Africa      32450
Name: numero_feridos, dtype: int64
```



Investigando os registros de feridos anualmente por ataques terroristas, percebe-se uma variação entre o pico de registros de feridos em cada região. A região

do Oriente Médio e Norte da África registrou o seu maior número de feridos no ano de 2016, com 23.994 registros. A região sul da Ásia totalizou em 2013 quase 11.100 registros, o seu maior número. A região da África subsaariana teve seu pico de registros em 2015, somando 6.395 feridos. A região sudeste da Ásia registrou em 2012 o seu maior número de feridos, contabilizados em 1.527 feridos. A região leste do continente asiático registrou seu pico em 2014, com 478 feridos registrados. Já a região central da Ásia somou 43 feridos no ano de 2010, o seu maior número no período pesquisado.



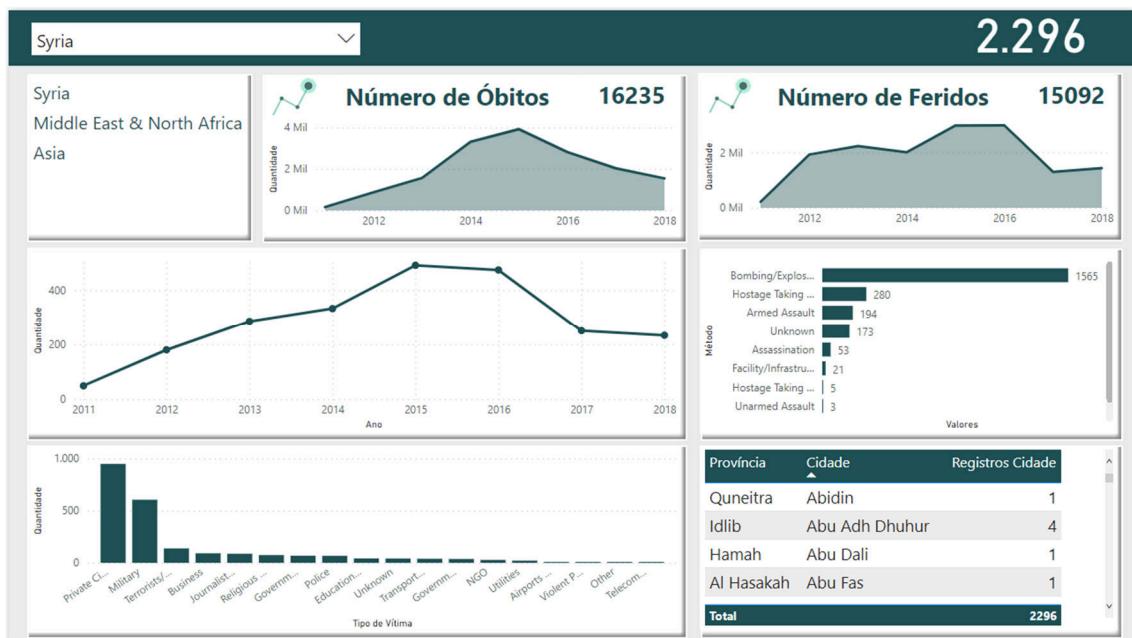
Ao final da exploração dos dados através do Jupyter Notebook, o *dataset* com as informações nulas atualizadas foi exportado para o desenvolvimento dos modelos de *machine learning*.

Para uma análise individual de cada território foi desenvolvido um *dashboard* utilizando o programa Microsoft Power BI.

Através do *dashboard* desenvolvido no Microsoft Power BI é possível observar algumas das informações de ataques terroristas individualmente para cada território. É possível verificar o número de registros totais de eventos no território, assim como o número de registros anuais.

No *dashboard* é possível observar o número de óbitos e feridos em cada território, tanto o número absoluto, quanto os registros anuais em todo o período estudado.

Informações como o número de registros para cada método utilizado durante os ataques, assim como o número de vítimas e alvos e as províncias e cidades onde há registros de ataques terroristas podem ser observadas no *dashboard*.



7. Criação de modelos de Machine Learning

A seção de criação dos modelos de *machine learning* está dividida, além desta introdução, em outras duas partes: o desenvolvimento a partir da base de dados de ataques terroristas e o desenvolvimento a partir da base de dados que combina indicadores sociais com a base de dados de ataques terroristas. Nesse primeiro momento serão definidos e explicados os algoritmos de classificação utilizados durante os testes para criação dos modelos.

Os algoritmos utilizados para os testes e desenvolvimento dos modelos foram: Árvore de Decisão, Random Forest e o KNN (K-Nearest Neighbors). Por motivos de comparação, os mesmos algoritmos foram utilizados para o desenvolvimento de ambos os modelos.

Inicialmente será feita uma breve explicação dos algoritmos utilizados e em seguida, serão mostrados os processos de desenvolvimento dos modelos.

Árvore de Decisão (*Decision Tree*) é um método de aprendizado supervisionado utilizado para classificação e regressão. Com uma estrutura semelhante a um fluxograma, onde os dados são continuamente divididos de acordo com certos parâmetros, a árvore de decisão pode ser desenhada de cima para baixo, com sua raiz no topo e ramificada em duas entidades que representam os valores resultantes de testes em um atributo: os nós de decisão e as folhas. Os nós de decisão representam os testes em um atributo e sua consequente divisão, enquanto as folhas podem ser consideradas parte de decisões dessas divisões ou uma decisão final. Entre as principais vantagens do algoritmo de Árvore de Decisão destacam-se a facilidade para se interpretar e entender resultados, sua capacidade de trabalhar com variáveis contínuas e categóricas, além de fornecer uma clara indicação dos atributos mais importantes para previsões e classificações. Entretanto, entre as principais desvantagens do algoritmo destaca-se a instabilidade para se reproduzir uma árvore, já que uma pequena alteração nos dados pode causar uma grande mudança em sua estrutura.

O algoritmo Random Forest consiste em um grande número de árvores de decisão individuais, onde cada árvore é construída aleatoriamente e constituída por diferentes amostras de dados aleatórios extraídos de um conjunto de treinamento através de um processo de substituição. Assim, a floresta consiste em árvores de decisão ligeiramente diferentes e com pequenas variações entre si, onde a variável categórica mais frequente produzirá a classe prevista, fazendo com o algoritmo apresente maior estabilidade. Dessa forma, o algoritmo é capaz de evitar o super justiça ou *overfitting*. Porém, devido aos processos para construção de árvores individuais, o algoritmo Random Forest pode apresentar certa complexidade e um tempo maior de treinamento.

O algoritmo k-nearest neighbor ou K-NN é um algoritmo baseado em técnica de aprendizado de máquina que armazena todos os dados disponíveis e classifica um novo registro com base na similaridade (ou distância) entre registros já classificados.

Isto é, o K-NN apropria-se da ideia de similaridade entre os dados com a utilização de cálculos matemáticos para determinar a distância entre eles e classificá-los na categoria mais semelhante entre as categorias disponíveis. Entre os pontos positivos do algoritmo KNN é possível ressaltar a sua facilidade de implementação e a sua eficiência mesmo com uma base de treinamento com muitos dados. O principal ponto negativo do algoritmo KNN é o seu alto custo computacional devido aos cálculos de distância entre os pontos realizados para todas as amostras de treinamento.

Durante o desenvolvimento dos modelos foram apresentados valores de acurácia para ambas as bases de treinamento e teste, porém a análise dos resultados finais será feita pela métrica F1-score. A métrica de avaliação F1-score da macro média é a média harmônica de todas as pontuações F1 de cada classe.

Junto a acurácia, a análise das outras medidas de avaliação será feita através da média macro, já que a exploração da base de dados permite observar que se trata de uma base com múltiplas opções de classificação. Ainda que haja um desbalanceamento de classes, todas são igualmente importantes, por isso, a escolha pela análise da macro média.

Além da acurácia, foram utilizados outros dois elementos para auxiliar na visualização de análise de desempenho dos modelos: a Matriz de Confusão, que mostra o número de classificações corretas e o relatório de classificação.

7.1. Criação de modelos de Machine Learning com a base de dados de ataques terroristas

Iniciando o processo de desenvolvimento dos modelos de *machine learning* utilizando apenas a base de dados de ataques terroristas, os processos para desenvolvimento dos modelos seguem o mesmo padrão após análise e transformação dos dados.

Inicialmente, as bibliotecas são carregadas.

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt

from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score, confusion_matrix, classification_report
from yellowbrick.classifier import ClassificationReport
```

Em seguida, o arquivo de dados de ataques terroristas é importado. O arquivo para desenvolvimento dos modelos de machine learning foi obtido após o tratamento e exploração dos dados de ataques terroristas através do Jupyter Notebook.

```
avaliacao_terrorismo = pd.read_csv("C:/Users/Spaox/Documents/Projeto/Arquivos/Tabelas/Exportadas/analise_terrorismo.csv", sep=";")
```

| | id | id_evento | ano | mes | dia | nome_continente | codigo_pais | nome_pais | iso | codigo_regiao | ... | grupo_terroris |
|-------|-------|--------------|------|-----|-----|-----------------|-------------|-------------|-----|---------------|-----|-----------------------------------|
| 0 | 1 | 201001010002 | 2010 | 1 | 1 | Asia | 4 | Afghanistan | AF | 6 | ... | Talibã |
| 1 | 2 | 201001010003 | 2010 | 1 | 1 | Asia | 153 | Pakistan | PK | 6 | ... | Tehrik-i-Taliba Pakistan (TTP) |
| 2 | 3 | 201001010004 | 2010 | 1 | 1 | Asia | 153 | Pakistan | PK | 6 | ... | Unknov |
| 3 | 4 | 201001010005 | 2010 | 1 | 1 | Asia | 153 | Pakistan | PK | 6 | ... | Unknov |
| 4 | 5 | 201001010006 | 2010 | 1 | 1 | Asia | 153 | Pakistan | PK | 6 | ... | Unknov |
| ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| 90750 | 90751 | 201812310029 | 2018 | 12 | 31 | Asia | 4 | Afghanistan | AF | 6 | ... | Unknov |
| 90751 | 90752 | 201812310030 | 2018 | 12 | 31 | Asia | 4 | Afghanistan | AF | 6 | ... | Talibã |
| 90752 | 90753 | 201812310031 | 2018 | 12 | 31 | Asia | 4 | Afghanistan | AF | 6 | ... | Talibã |
| 90753 | 90754 | 201812310032 | 2018 | 12 | 31 | Asia | 4 | Afghanistan | AF | 6 | ... | Talibã |
| 90754 | 90755 | 201812310033 | 2018 | 12 | 31 | Asia | 4 | Afghanistan | AF | 6 | ... | Talibã |

90755 rows × 38 columns

É realizada uma breve leitura dos dados para escolha dos atributos previsores e do atributo classe que serão utilizados no desenvolvimento dos modelos. Para isso, é criado um novo *dataset* com os atributos escolhidos.

Os atributos previsores determinados foram: ano, mês e dia do ataque, primeiro critério para determinação de um ataque como evento terrorista, segundo critério para determinação de um ataque como evento terrorista, terceiro critério para determinação de um ataque como evento terrorista, se o evento foi conectado a outros ou não, se houve sucesso no ataque, método de ataque, tipo de vítima/alvo, nacionalidade da vítima/alvo, se houve identificação do grupo perpetrador, se houve reivindicação do ataque, tipo de arma utilizada, número de óbitos decorrentes do ataque, número de feridos decorrentes do ataque, se houve danos em propriedades e dúvida se foi terrorismo.

```
dataset_avaliacao_terrorismo = avaliacao_terrorismo[['nome_regiao', 'ano', 'mes', 'dia', 'criterio1','criterio2','criterio3', 'at
```

| dataset_avaliacao_terrorismo | | | | | | | | | | | | | |
|------------------------------|-------------|------|-----|-----|-----------|-----------|-----------|--------------------|----------------|--------------------------------|-----------------------------|-------------|-------------|
| | nome_regiao | ano | mes | dia | criterio1 | criterio2 | criterio3 | ataques_conectados | sucesso_ataque | metodo_ataque | tipo_vitima | nac_alvo | identificac |
| 0 | South Asia | 2010 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | Bombing/Explosion | Private Citizens & Property | Afghanistan | |
| 1 | South Asia | 2010 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | Bombing/Explosion | Private Citizens & Property | Pakistan | |
| 2 | South Asia | 2010 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | Bombing/Explosion | Private Citizens & Property | Pakistan | |
| 3 | South Asia | 2010 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | Bombing/Explosion | Educational Institution | Pakistan | |
| 4 | South Asia | 2010 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | Bombing/Explosion | Educational Institution | Pakistan | |
| ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| 90750 | South Asia | 2018 | 12 | 31 | 1 | 1 | 1 | 0 | 1 | Armed Assault | Government (General) | Afghanistan | |
| 90751 | South Asia | 2018 | 12 | 31 | 1 | 1 | 1 | 0 | 1 | Facility/Infrastructure Attack | Police | Afghanistan | |
| 90752 | South Asia | 2018 | 12 | 31 | 1 | 1 | 0 | 0 | 1 | Facility/Infrastructure Attack | Military | Afghanistan | |
| 90753 | South Asia | 2018 | 12 | 31 | 1 | 1 | 1 | 0 | 1 | Unknown | Private Citizens & Property | Afghanistan | |
| 90754 | South Asia | 2018 | 12 | 31 | 1 | 1 | 1 | 0 | 1 | Unknown | Police | Afghanistan | |

90755 rows x 19 columns

São realizadas então duas verificações no novo *dataset*: verificação de quantidade de registros nulos.

```
dataset_avaliacao_terrorismo.isnull().sum()
```

| | |
|----------------------|---|
| nome_regiao | 0 |
| ano | 0 |
| mes | 0 |
| dia | 0 |
| criterio1 | 0 |
| criterio2 | 0 |
| criterio3 | 0 |
| ataques_conectados | 0 |
| sucesso_ataque | 0 |
| metodo_ataque | 0 |
| tipo_vitima | 0 |
| nac_alvo | 0 |
| identificacao_grupo | 0 |
| reivindicacao_ataque | 0 |
| tipo_arma | 0 |
| numero_obitos | 0 |
| numero_feridos | 0 |
| danos_propriedades | 0 |
| dvida_terrorismo | 0 |
| dtype: int64 | |

E tipo de registros por coluna.

```
dataset_avaliacao_terrorismo.info()
```

| # | Column | Non-Null Count | Dtype |
|----|------------------------------|----------------|-----------------|
| 0 | nome_regiao | 90755 | non-null object |
| 1 | ano | 90755 | non-null int64 |
| 2 | mes | 90755 | non-null int64 |
| 3 | dia | 90755 | non-null int64 |
| 4 | criterio1 | 90755 | non-null int64 |
| 5 | criterio2 | 90755 | non-null int64 |
| 6 | criterio3 | 90755 | non-null int64 |
| 7 | ataques_conectados | 90755 | non-null int64 |
| 8 | sucesso_ataque | 90755 | non-null int64 |
| 9 | metodo_ataque | 90755 | non-null object |
| 10 | tipo_vitima | 90755 | non-null object |
| 11 | nac_alvo | 90755 | non-null object |
| 12 | identificacao_grupo | 90755 | non-null int64 |
| 13 | reivindicacao_ataque | 90755 | non-null int64 |
| 14 | tipo_arma | 90755 | non-null object |
| 15 | numero_obitos | 90755 | non-null int64 |
| 16 | numero_feridos | 90755 | non-null int64 |
| 17 | danos_propriedades | 90755 | non-null int64 |
| 18 | dvida_terrorismo | 90755 | non-null int64 |
| | dtypes: int64(14), object(5) | | |
| | memory usage: 13.2+ MB | | |

No prosseguimento são definidas as variáveis para os atributos previsores e o atributo classe.

```
X_avaliacao = dataset_avaliacao_terrorismo.iloc[:, 1:19].values
X_avaliacao

array([[2010, 1, 1, ..., 0, 1, 0],
       [2010, 1, 1, ..., 0, 1, 0],
       [2010, 1, 1, ..., 0, 0, 0],
       ...,
       [2018, 12, 31, ..., 0, 1, 1],
       [2018, 12, 31, ..., 0, 0, 0],
       [2018, 12, 31, ..., 3, 0, 0]], dtype=object)

X_avaliacao[0]

array([2010, 1, 1, 1, 1, 0, 1, 'Bombing/Explosion',
      'Private Citizens & Property', 'Afghanistan', 1, 1, 'Explosives',
      4, 0, 1, 0], dtype=object)
```

Criação da variável para a classe *nome_regiao*

```
y_avaliacao = dataset_avaliacao_terrorismo.iloc[:, 0].values
y_avaliacao

array(['South Asia', 'South Asia', 'South Asia', ...,
       'South Asia', 'South Asia'], dtype=object)

y_avaliacao[0]

'South Asia'
```

Em seguida, as variáveis categóricas de atributos previsores são convertidas para valores numéricos através da ferramenta *Label Encoder*.

```
from sklearn.preprocessing import LabelEncoder

label_encoder_metodo_ataque = LabelEncoder()
label_encoder_tipo_vitima = LabelEncoder()
label_encoder_nac_alvo = LabelEncoder()
label_encoder_tipo_arma = LabelEncoder()

X_avaliacao[:,8] = label_encoder_metodo_ataque.fit_transform(X_avaliacao[:,8])
X_avaliacao[:,9] = label_encoder_tipo_vitima.fit_transform(X_avaliacao[:,9])
X_avaliacao[:,10] = label_encoder_nac_alvo.fit_transform(X_avaliacao[:,10])
X_avaliacao[:,13] = label_encoder_tipo_arma.fit_transform(X_avaliacao[:,13])

X_avaliacao[0]

array([2010, 1, 1, 1, 1, 0, 1, 2, 12, 0, 1, 1, 2, 4, 0, 1, 0],
      dtype=object)

X_avaliacao

array([[2010, 1, 1, ..., 0, 1, 0],
       [2010, 1, 1, ..., 0, 1, 0],
       [2010, 1, 1, ..., 0, 0, 0],
       ...,
       [2018, 12, 31, ..., 0, 1, 1],
       [2018, 12, 31, ..., 0, 0, 0],
       [2018, 12, 31, ..., 3, 0, 0]], dtype=object)
```

Ao fim do processo de conversão, as variáveis são dispostas em um novo *data frame*.

```
datafx_avaliacao = pd.DataFrame(X_avaliacao)
```

Em seguida, o *data frame* com as novas variáveis é exportado e novamente carregado.

```
datafx_avaliacao.to_csv(r"C:\Users\Spaox\Documents\Projeto\Arquivos\Tabelas\Exportadas\datafx_avaliacao.csv")
datafx_avaliacao = pd.read_csv(r"C:\Users\Spaox\Documents\Projeto\Arquivos\Tabelas\Exportadas\datafx_avaliacao.csv")
```

Assim é possível observar que as variáveis agora são numéricas

```
datafx_avaliacao.info()
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 98755 entries, 0 to 98754
Data columns (total 18 columns):
 #   Column   Non-Null Count  Dtype  
--- 
 0   0          98755 non-null   int64  
 1   1          98755 non-null   int64  
 2   2          98755 non-null   int64  
 3   3          98755 non-null   int64  
 4   4          98755 non-null   int64  
 5   5          98755 non-null   int64  
 6   6          98755 non-null   int64  
 7   7          98755 non-null   int64  
 8   8          98755 non-null   int64  
 9   9          98755 non-null   int64  
 10  10         98755 non-null   int64  
 11  11         98755 non-null   int64  
 12  12         98755 non-null   int64  
 13  13         98755 non-null   int64  
 14  14         98755 non-null   int64  
 15  15         98755 non-null   int64  
 16  16         98755 non-null   int64  
 17  17         98755 non-null   int64  
dtypes: int64(18)
memory usage: 12.5 MB
```

Para o processo de escalonamento dos valores é utilizada a ferramenta *Standard Scaler*. Através dessa biblioteca é possível deixar todos os atributos na mesma escala e permitir uma padronização. É importando o uso dessa biblioteca para que os valores não tenham “pesos” diferentes durante o treinamento dos modelos.

Assim, os valores dos atributos previsores foram padronizados.

```
from sklearn.preprocessing import StandardScaler

avaliacao_scaler = StandardScaler()
x_avaliacao_scaler = avaliacao_scaler.fit_transform(X_avaliacao)

x_avaliacao_scaler[0]
array([-2.08387111, -1.60681313, -1.67157021,  0.09519242,  0.07322317,
       0.42800083, -0.46110872,  0.40923966, -0.0953298 ,  0.45774342,
      -1.74682014,  0.9847887 ,  2.06455208, -0.68752547,  0.14811501,
     -0.25219069,  0.62604505, -0.46170595])
```

Após o escalonamento dos valores, as variáveis dos atributos previsores e atributo classe foram separadas em bases de teste e treinamento através da biblioteca *train_test_split* do *Scikit Learn*.

Foram assim separados 75% dos registros para a base de treinamento e 25% para a base de teste. Através do parâmetro *random_state* é possível gerar a base de dados com os mesmos registros para comparar teste e treinamento.

```
X_avaliacao_treinamento, X_avaliacao_teste, y_avaliacao_treinamento, y_avaliacao_teste = train_test_split(x_avaliacao_scaler, y_avaliacao, test_size=0.25, random_state=42)

X_avaliacao_teste.shape, y_avaliacao_teste.shape
((22689, 18), (22689,))

X_avaliacao_treinamento.shape, y_avaliacao_treinamento.shape
((68066, 18), (68066,))
```

Após a divisão dos registros em bases de teste e treinamento, deu-se início ao desenvolvimento dos modelos e utilização dos algoritmos de *machine learning*.

Inicialmente foi realizado a importação do algoritmo de Árvore de Decisão (*Decision Tree Classifier*) e iniciado o processo de ajuste das bases.

```
from sklearn.tree import DecisionTreeClassifier

dtc_avaliacao = DecisionTreeClassifier()
dtc_avaliacao.fit(X_avaliacao_treinamento, y_avaliacao_treinamento)

#previsoes
dtc_avaliacao_yteste      = dtc_avaliacao.predict(X_avaliacao_teste)      #previsores de teste
dtc_avaliacao_ytreinamento = dtc_avaliacao.predict(X_avaliacao_treinamento)#previsores de treinamento

print('Acurácia dos dados de Treinamento: {}'.format(accuracy_score(y_avaliacao_treinamento, dtc_avaliacao_ytreinamento)))
print('Acurácia dos dados de Teste: {}'.format(accuracy_score(y_avaliacao_teste, dtc_avaliacao_yteste)))

Acurácia dos dados de Treinamento: 0.999926541885168
Acurácia dos dados de Teste: 0.9802106747763233
```

Ao obtermos os resultados da acurácia do algoritmo, pode-se observar o valor de 99% de acurácia para os dados de treinamento e 98% para os dados de teste.

Em seguida, foi realizada o processo de *tunning* dos parâmetros, isto é, encontrar os melhores hiperparâmetros para o modelo. Para isso, foi utilizado o GridSearchCV. Através dessa ferramenta é realizada a pesquisa em grade dos melhores parâmetros através da validação cruzada.

```
from sklearn.model_selection import GridSearchCV

parametros = {'criterion': ['gini', 'entropy'],
              'splitter': ['best', 'random'],
              # Divisão para cada um dos nós
              # Teste do 'best' como default e random
              'min_samples_split': [2, 5, 10, 15],
              # Número mínimo de registros requerido para dividir uma árvore de decisão
              'min_samples_leaf': [1, 3, 5, 10, 15],
              # Número mínimo de registros requeridos para um nó ser reconhecido como um nó folha
              'max_depth': [None, 2, 3, 5, 7, 10]}

grid_search = GridSearchCV(estimator=DecisionTreeClassifier(), param_grid=parametros, cv=5)
# GridSearch combinará todos os parâmetros testando combinados um por um
grid_search.fit(X_avaliacao_treinamento, y_avaliacao_treinamento)
# X -> Atributos previsores
# y -> Respostas esperadas
# fit -> Ajuste e treinamento
melhores_parametros = grid_search.best_params_
# Melhor combinação de parâmetros
melhor_resultado = grid_search.best_score_
# Resultados
print('Melhores Parâmetros: ' + str(melhores_parametros))
print('Melhor Resultado: ' + str (melhor_resultado))

Melhores Parâmetros: {'criterion': 'entropy', 'max_depth': None, 'min_samples_leaf': 10, 'min_samples_split': 2, 'splitter': 'best'}
Melhor Resultado: 0.983692295399716
```

Após obter os melhores parâmetros através do GridSearchCV, o algoritmo é novamente executado, utilizando os novos parâmetros selecionados. Os hiperparâmetros encontrados foram: criterion: 'entropy', max_depth: 'None', min_samples_leaf: 10, min_samples_split: 2, splitter: 'best'

```
dtc_avaliacao = DecisionTreeClassifier(criterion='entropy', max_depth=None, min_samples_leaf=10, min_samples_split=2, splitter='best')
dtc_avaliacao.fit(X_avaliacao_treinamento, y_avaliacao_treinamento)
dtc_avaliacao_yteste      = dtc_avaliacao.predict(X_avaliacao_teste)      #previsores de teste
dtc_avaliacao_ytreinamento = dtc_avaliacao.predict(X_avaliacao_treinamento)
print('Acurácia de Teste Após Tunning dos Parâmetros: {}'.format(accuracy_score(y_avaliacao_teste, dtc_avaliacao_yteste)))
```

É possível observar que após o *tunning* dos parâmetros houve uma pequena melhora na acurácia do algoritmo em comparação ao processo realizado previamente

a pesquisa em grade. Após a execução do algoritmo com os hiper parâmetros selecionados a acurácia foi de 98.28%.

Através da Matriz de Confusão é possível visualizar o desempenho de classificação do algoritmo após o *tunning* dos parâmetros.



Com o relatório de classificação é possível analisar medidas de avaliação do algoritmo.

```
print(classification_report(y_avalicao_teste, dtc_avalicao_yteste))

precision    recall  f1-score   support

Central Asia      0.68      0.59      0.63       29
East Asia         0.63      0.68      0.66       28
Middle East & North Africa  0.99      0.99      0.99     9143
South Asia        0.98      0.99      0.99      8123
Southeast Asia    0.99      0.99      0.99     1944
Sub-Saharan Africa 0.96      0.97      0.96     3422

accuracy                           0.98      22689
macro avg       0.87      0.87      0.87      22689
weighted avg     0.98      0.98      0.98      22689
```

O segundo algoritmo a ser testado foi o Random Forest. Inicialmente foi feita a importação do algoritmo e o processo de ajuste das bases para obtenção da acurácia.

```

from sklearn.ensemble import RandomForestClassifier

rfc_avaliacao = RandomForestClassifier()
rfc_avaliacao.fit(X_avaliacao_treinamento, y_avaliacao_treinamento)

#previsões
rfc_avaliacao_yteste = rfc_avaliacao.predict(X_avaliacao_teste)      #previsores de teste
rfc_avaliacao_ytreinamento = rfc_avaliacao.predict(X_avaliacao_treinamento)#previsores de treinamento

```

Assim, ao fim do processo pode-se obter a acurácia de teste e treinamento do algoritmo.

```

print('Acurácia dos dados de Treinamento: {}'.format(accuracy_score(y_avaliacao_treinamento, rfc_avaliacao_ytreinamento)))
print('Acurácia dos dados de Teste: {}'.format(accuracy_score(y_avaliacao_teste, rfc_avaliacao_yteste)))

Acurácia dos dados de Treinamento: 0.9999265418858168
Acurácia dos dados de Teste: 0.9661069240601172

```

Os resultados da acurácia do algoritmo mostram o valor de 99% de acurácia para os dados de treinamento e 96.61% para os dados de teste.

Em seguida, foi realizada a busca em grade para os hiper parâmetros do algoritmo.

```

parametros_rfc = {'criterion': ['gini', 'entropy'],
                  # Método de divisão
                  'n_estimators': [10, 40, 100, 150],
                  # Definir número de árvores
                  'min_samples_split': [2, 5, 10],
                  # Número mínimo de registros requerido para dividir uma árvore de decisão
                  'min_samples_leaf': [1, 5, 10]}
                  # Número mínimo de registros requeridos para um nó ser reconhecido como um nó folha

grid_search = GridSearchCV(estimator=RandomForestClassifier(), param_grid=parametros_rfc, cv=5)
grid_search.fit(X_avaliacao_treinamento, y_avaliacao_treinamento)
melhores_parametros_rfc = grid_search.best_params_
melhor_resultado_rfc = grid_search.best_score_
print('Melhores Parâmetros: ' + str(melhores_parametros_rfc))
print('Melhor Resultado: ' + str(melhor_resultado_rfc))

Melhores Parâmetros: {'criterion': 'gini', 'min_samples_leaf': 1, 'min_samples_split': 5, 'n_estimators': 150}
Melhor Resultado: 0.9669202476080949

```

Após obter-se os melhores parâmetros através do Grid Search, o algoritmo é novamente executado, utilizando os novos parâmetros selecionados. O hiper parâmetros selecionados foram: criterion: 'gini', min_samples_leaf: 1, min_samples_split: 5, n_estimators: 150.

```

rfc_avaliacao = RandomForestClassifier(criterion='gini', min_samples_leaf=1, min_samples_split=5, n_estimators=150)
rfc_avaliacao.fit(X_avaliacao_treinamento, y_avaliacao_treinamento)

rfc_avaliacao_yteste = rfc_avaliacao.predict(X_avaliacao_teste)
rfc_avaliacao_ytreinamento = rfc_avaliacao.predict(X_avaliacao_treinamento)
print('Acurácia de Teste Após Tuning dos Parâmetros: {}'.format(accuracy_score(y_avaliacao_teste, rfc_avaliacao_yteste)))

Acurácia de Teste Após Tuning dos Parâmetros: 0.9652695138613425

```

Após o novo treinamento do algoritmo, houve uma pequena baixa na acurácia de teste, obtendo agora o valor de 96.52% de acurácia.

Através da Matriz de Confusão é possível visualizar o desempenho de classificação do algoritmo após a execução com os hiper parâmetros.



Através do relatório de classificação é possível fazer uma análise sobre a capacidade de classificação do algoritmo.

```
print(classification_report(y_avaliacao_teste, rfc_avaliacao_yteste))

precision    recall  f1-score   support

Central Asia      0.00      0.00      0.00       29
      East Asia     1.00      0.18      0.30       28
Middle East & North Africa  0.98      0.98      0.98     9143
      South Asia    0.97      0.99      0.98     8123
      Southeast Asia 0.97      0.92      0.94     1944
      Sub-Saharan Africa 0.93      0.92      0.93     3422

accuracy                           0.97    22689
macro avg       0.81      0.66      0.69    22689
weighted avg    0.96      0.97      0.96    22689
```

O próximo algoritmo a ser testado foi o KNN. Primeiramente foi feita a importação do algoritmo, assim como o processo de ajuste das bases.

```
from sklearn.neighbors import KNeighborsClassifier

knn_avaliacao = KNeighborsClassifier()
knn_avaliacao.fit(X_avaliacao_treinamento, y_avaliacao_treinamento)

#previsoes
knn_avaliacao_yteste = knn_avaliacao.predict(X_avaliacao_teste)      #previsores de teste
knn_avaliacao_ytreinamento = knn_avaliacao.predict(X_avaliacao_treinamento)#previsores de treinamento
```

Em seguida, obtemos o resultado da acurácia de treinamento e teste do algoritmo.

```
print('Acurácia dos dados de Treinamento: {}'.format(accuracy_score(y_avaliacao_treinamento, knn_avaliacao_ytreinamento)))
print('Acurácia dos dados de Teste: {}'.format(accuracy_score(y_avaliacao_teste, knn_avaliacao_yteste)))

Acurácia dos dados de Treinamento: 0.8064525607498604
Acurácia dos dados de Teste: 0.7269602009784477
```

O algoritmo teve 80% de acurácia para os dados de treinamento, enquanto para os dados de teste o valor foi de 72%.

Iniciou-se assim o processo de busca em grade pelos hiper parâmetros mais adequados para o algoritmo.

```
parametros_knn = {'n_neighbors': [3, 5, 10, 20],
                  'weights': ['uniform', 'distance'],
                  # Número de vizinhos para o cálculo de distância
                  'metric': ['euclidean', 'manhattan']}
                  # Cálculo para medir a similaridade dos parâmetros
                  # 1 -> Distância Manhattan
                  # 2 -> Distância Euclidiana

grid_search = GridSearchCV(estimator=KNeighborsClassifier(), param_grid=parametros_knn, cv=3)
grid_search.fit(X_avaliacao_treinamento, y_avaliacao_treinamento)
melhores_parametros_knn = grid_search.best_params_
melhor_resultado_knn = grid_search.best_score_
print('Melhores Parâmetros: ' + str(melhores_parametros_knn))
print('Melhor Resultado: ' + str(melhor_resultado_knn))

Melhores Parâmetros: {'metric': 'manhattan', 'n_neighbors': 10, 'weights': 'distance'}
Melhor Resultado: 0.7745276730781914
```

E assim, os hiper parâmetros obtidos foram: metric: ‘manhattan’, n_neighbors: 10, weights: ‘distance’.

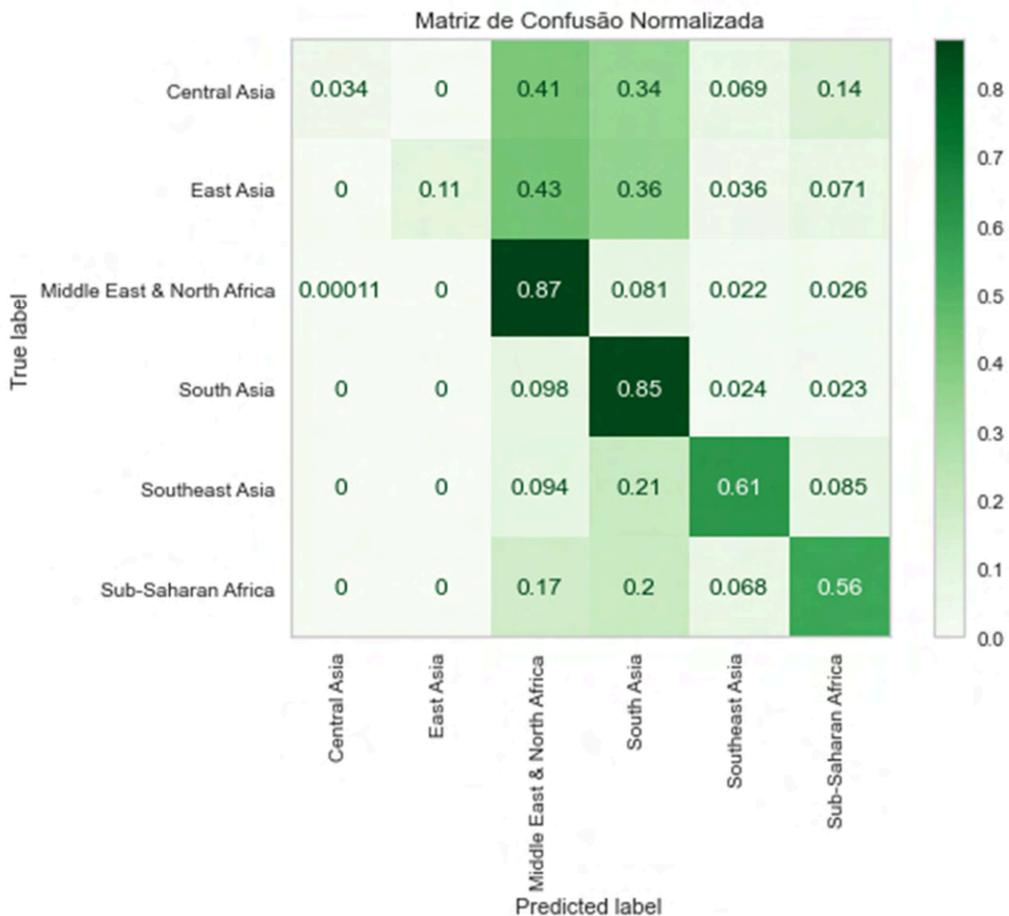
Em seguida, o algoritmo foi novamente treinado, dessa vez com novos hiper parâmetros selecionados anteriormente.

```
knn_avaliacao = KNeighborsClassifier(metric='manhattan', n_neighbors=10, weights='distance')
knn_avaliacao.fit(X_avaliacao_treinamento, y_avaliacao_treinamento)

knn_avaliacao_yteste      = knn_avaliacao.predict(X_avaliacao_teste)      #previsores de teste
knn_avaliacao_ytreinamento = knn_avaliacao.predict(X_avaliacao_treinamento)#previsores de treinamento
teste = print('Acurácia de Teste Após Tuning dos Parâmetros: {}'.format(accuracy_score(y_avaliacao_teste, knn_avaliacao_yteste)))
Acurácia de Teste Após Tuning dos Parâmetros: 0.793776719996474
```

Pode-se observar que houve uma grande melhora na acurácia do algoritmo após a seleção dos hiper parâmetros, onde a acurácia atinge agora 79%.

Foram desenvolvidos modelos de visualização como a Matriz de Confusão e o relatório de classificação para análise de desempenho de classificação do algoritmo.



```
print(classification_report(y_avaliacao_teste, knn_avaliacao_yteste))

precision    recall  f1-score   support

Central Asia      0.50     0.03    0.06      29
East Asia         1.00     0.11    0.19      28
Middle East & North Africa  0.83     0.87    0.85    9143
South Asia        0.79     0.85    0.82    8123
Southeast Asia    0.65     0.61    0.63    1944
Sub-Saharan Africa 0.76     0.56    0.64    3422

accuracy                           0.79    22689
macro avg       0.76     0.51    0.53    22689
weighted avg     0.79     0.79    0.79    22689
```

7.2. Criação de modelos de Machine Learning com a base de indicadores políticos e sociais em conjunto com registros de ataques terroristas

Nesta seção será tratada o desenvolvimento dos modelos de *machine learning* utilizando a base de dados de indicadores políticos e sociais em conjunto com a base de dados de ataques terroristas. Assim como descrito na seção anterior, os processos para desenvolvimento dos modelos seguem o mesmo padrão após a análise e transformação dos dados.

Primeiramente, são carregadas as bibliotecas.

```

import pandas as pd
import numpy as np
import matplotlib.pyplot as plt

from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score, confusion_matrix, classification_report
from yellowbrick.classifier import ClassificationReport

```

Em seguida, o arquivo de dados contendo indicadores sociais e os registros de ataques terroristas é importado.

```

avaliacao_indicadores = pd.read_csv("C:/Users/Spaox/Documents/Projeto/Arquivos/Tabelas/Exportadas/terrorismo_indicadores.csv", sep=';')
avaliacao_indicadores

```

| | id | id_evento | ano | mes | dia | nome_continente | codigo_pais | nome_pais | iso | codigo_regiao | ... | faixa_idh | indice_idh_regiao | indice_idh_continente |
|-------|-------|-----------|------|-----|-----|-----------------|-------------|-------------|-----|---------------|-----|-----------|-------------------|-----------------------|
| 0 | 89706 | 2,02E+11 | 2018 | 11 | 12 | Asia | 4 | Afghanistan | AF | 6 | ... | Low | Below Average | Below Average |
| 1 | 1868 | 2,01E+11 | 2010 | 6 | 20 | Asia | 4 | Afghanistan | AF | 6 | ... | Low | Below Average | Below Average |
| 2 | 82573 | 2,02E+11 | 2018 | 2 | 5 | Asia | 4 | Afghanistan | AF | 6 | ... | Low | Below Average | Below Average |
| 3 | 12621 | 2,01E+11 | 2012 | 6 | 16 | Asia | 4 | Afghanistan | AF | 6 | ... | Low | Below Average | Below Average |
| 4 | 89748 | 2,02E+11 | 2018 | 11 | 15 | Asia | 4 | Afghanistan | AF | 6 | ... | Low | Below Average | Below Average |
| ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| 90750 | 77189 | 2,02E+11 | 2017 | 7 | 19 | Africa | 231 | Zimbabwe | ZW | 11 | ... | Medium | Above Average | Above Average |
| 90751 | 18170 | 2,01E+11 | 2013 | 2 | 23 | Africa | 231 | Zimbabwe | ZW | 11 | ... | Low | Above Average | Above Average |
| 90752 | 76981 | 2,02E+11 | 2017 | 7 | 12 | Africa | 231 | Zimbabwe | ZW | 11 | ... | Medium | Above Average | Above Average |
| 90753 | 77198 | 2,02E+11 | 2017 | 7 | 19 | Africa | 231 | Zimbabwe | ZW | 11 | ... | Medium | Above Average | Above Average |
| 90754 | 18547 | 2,01E+11 | 2013 | 3 | 11 | Africa | 231 | Zimbabwe | ZW | 11 | ... | Low | Above Average | Above Average |

90755 rows x 66 columns

Após a importação é realizada uma leitura para escolha dos atributos previsores e do atributo classe que serão utilizados no desenvolvimento dos modelos. Para isso, é criado um novo *dataset* com os atributos escolhidos.

Os atributos previsores determinados foram: regime político no período, índice de fracialização étnica (Alto ou Baixo), religião predominante, religião secundária, índice de desemprego comparado a média da região, índice de educação comparado a média da região, índice de expectativa de vida comparado a média da região, classificação do índice de desenvolvimento humano, índice de desenvolvimento humano comparado a média da região, índice de controle de corrupção, índice de efetividade do governo, índice de estabilidade política, ano, mês e dia do ataque.

```

dataset_avaliacao_indicadores = avaliacao_indicadores[['nome_regiao', 'regime_politico','indice_divisao_etnica','religiao_predomini

```

| dataset_avaliacao_indicadores | | | | | | | | |
|-------------------------------|--------------------|---------------------|-----------------------|-----------------------|---------------------|--------------------------|------------------------|------------|
| # | nome_regiao | regime_politico | indice_divisao_etnica | religiao_predominante | religiao_secundaria | indice_desemprego_regiao | indice_educacao_regiao | indice_idh |
| 0 | South Asia | electoral autocracy | High Index | Muslims | Christians | Above Average | Below Average | 0.500000 |
| 1 | South Asia | electoral autocracy | High Index | Muslims | Christians | Above Average | Below Average | 0.500000 |
| 2 | South Asia | electoral autocracy | High Index | Muslims | Christians | Above Average | Below Average | 0.500000 |
| 3 | South Asia | electoral autocracy | High Index | Muslims | Christians | Above Average | Below Average | 0.500000 |
| 4 | South Asia | electoral autocracy | High Index | Muslims | Christians | Above Average | Below Average | 0.500000 |
| ... | ... | ... | ... | ... | ... | ... | ... | ... |
| 90750 | Sub-Saharan Africa | electoral autocracy | Low Index | Christians | Folk Religions | Below Average | Above Average | 0.490000 |
| 90751 | Sub-Saharan Africa | electoral autocracy | Low Index | Christians | Folk Religions | Below Average | Above Average | 0.490000 |
| 90752 | Sub-Saharan Africa | electoral autocracy | Low Index | Christians | Folk Religions | Below Average | Above Average | 0.490000 |
| 90753 | Sub-Saharan Africa | electoral autocracy | Low Index | Christians | Folk Religions | Below Average | Above Average | 0.490000 |
| 90754 | Sub-Saharan Africa | electoral autocracy | Low Index | Christians | Folk Religions | Below Average | Above Average | 0.490000 |

90755 rows × 16 columns

São realizadas então duas verificações no novo *dataset*: verificação de quantidade de registros nulos.

```
dataset_avaliacao_indicadores.isnull().sum()
```

| nome_regiao | 0 |
|--------------------------|---|
| regime_politico | 0 |
| indice_divisao_etnica | 0 |
| religiao_predominante | 0 |
| religiao_secundaria | 0 |
| indice_desemprego_regiao | 0 |
| indice_educacao_regiao | 0 |
| indice_expvida_regiao | 0 |
| faixa_idh | 0 |
| indice_idh_regiao | 0 |
| indice_ctrlcorrupcao | 0 |
| indice_efetividadegov | 0 |
| indice_estpolitica | 0 |
| ano | 0 |
| mes | 0 |
| dia | 0 |

E tipo de registros por coluna.

```
dataset_avaliacao_indicadores.info()
```

| # | Column | Non-Null Count | Dtype |
|----|--------------------------|----------------|-----------------|
| 0 | nome_regiao | 90755 | non-null object |
| 1 | regime_politico | 90755 | non-null object |
| 2 | indice_divisao_etnica | 90755 | non-null object |
| 3 | religiao_predominante | 90755 | non-null object |
| 4 | religiao_secundaria | 90755 | non-null object |
| 5 | indice_desemprego_regiao | 90755 | non-null object |
| 6 | indice_educacao_regiao | 90755 | non-null object |
| 7 | indice_expvida_regiao | 90755 | non-null object |
| 8 | faixa_idh | 90755 | non-null object |
| 9 | indice_idh_regiao | 90755 | non-null object |
| 10 | indice_ctrlcorrupcao | 90755 | non-null object |
| 11 | indice_efetividadegov | 90755 | non-null object |
| 12 | indice_estpolitica | 90755 | non-null object |
| 13 | ano | 90755 | non-null int64 |
| 14 | mes | 90755 | non-null int64 |
| 15 | dia | 90755 | non-null int64 |

No prosseguimento são definidas as variáveis para os atributos previsores e o atributo classe.

A variável *X_avaliacao* contém os atributos previsores.

```
X_avaliacao = dataset_avaliacao_indicadores.iloc[:, 1:16].values
X_avaliacao

array([['electoral autocracy', 'High Index', 'Muslims', ..., 2018, 11,
       12],
       ['electoral autocracy', 'High Index', 'Muslims', ..., 2010, 6, 20],
       ['electoral autocracy', 'High Index', 'Muslims', ..., 2018, 2, 5],
       ...,
       ['electoral autocracy', 'Low Index', 'Christians', ..., 2017, 7,
        12],
       ['electoral autocracy', 'Low Index', 'Christians', ..., 2017, 7,
        19],
       ['electoral autocracy', 'Low Index', 'Christians', ..., 2013, 3,
        11]], dtype=object)

X_avaliacao[0]

array(['electoral autocracy', 'High Index', 'Muslims', 'Christians',
       'Above Average', 'Below Average', 'Below Average', 'Low',
       'Below Average', 'Low Rank', 'Low Rank', 'Low Rank', 2018, 11, 12],
      dtype=object)
```

A variável *y_avaliacao* contém o atributo classe.

```
y_avaliacao = dataset_avaliacao_indicadores.iloc[:, 0].values
y_avaliacao

array(['South Asia', 'South Asia', 'South Asia', ...,
       'Sub-Saharan Africa', 'Sub-Saharan Africa', 'Sub-Saharan Africa'],
      dtype=object)

y_avaliacao[0]

'South Asia'
```

Em seguida, as variáveis categóricas de atributos previsores são convertidas para valores numéricos através da ferramenta *Label Encoder*.

```
from sklearn.preprocessing import LabelEncoder

label_encoder_regime_politico = LabelEncoder()
label_encoder_indice_divisao_etnica = LabelEncoder()
label_encoder_religiao_predominante = LabelEncoder()
label_encoder_religiao_secundaria = LabelEncoder()
label_encoder_indice_desemprego_regiao = LabelEncoder()
label_encoder_indice_educacao_regiao = LabelEncoder()
label_encoder_indice_expvida_regiao = LabelEncoder()
label_encoder_faixa_idh = LabelEncoder()
label_encoder_indice_idh_regiao = LabelEncoder()
label_encoder_indice_ctrlcorrupcao = LabelEncoder()
label_encoder_indice_efetividadegov = LabelEncoder()
label_encoder_indice_estpolitica = LabelEncoder()

X_avaliacao[:,0] = label_encoder_regime_politico.fit_transform(X_avaliacao[:,0])
X_avaliacao[:,1] = label_encoder_indice_divisao_etnica.fit_transform(X_avaliacao[:,1])
X_avaliacao[:,2] = label_encoder_religiao_predominante.fit_transform(X_avaliacao[:,2])
X_avaliacao[:,3] = label_encoder_religiao_secundaria.fit_transform(X_avaliacao[:,3])
X_avaliacao[:,4] = label_encoder_indice_desemprego_regiao.fit_transform(X_avaliacao[:,4])
X_avaliacao[:,5] = label_encoder_indice_educacao_regiao.fit_transform(X_avaliacao[:,5])
X_avaliacao[:,6] = label_encoder_indice_expvida_regiao.fit_transform(X_avaliacao[:,6])
X_avaliacao[:,7] = label_encoder_faixa_idh.fit_transform(X_avaliacao[:,7])
X_avaliacao[:,8] = label_encoder_indice_idh_regiao.fit_transform(X_avaliacao[:,8])
X_avaliacao[:,9] = label_encoder_indice_ctrlcorrupcao.fit_transform(X_avaliacao[:,9])
X_avaliacao[:,10] = label_encoder_indice_efetividadegov.fit_transform(X_avaliacao[:,10])
X_avaliacao[:,11] = label_encoder_indice_estpolitica.fit_transform(X_avaliacao[:,11])

X_avaliacao

array([[1, 0, 6, ..., 2018, 11, 12],
       [1, 0, 6, ..., 2010, 6, 20],
       [1, 0, 6, ..., 2018, 2, 5],
       ...,
       [1, 1, 2, ..., 2017, 7, 12],
       [1, 1, 2, ..., 2017, 7, 19],
       [1, 1, 2, ..., 2013, 3, 11]], dtype=object)

X_avaliacao[0]

array([1, 0, 6, 1, 0, 2, 2, 2, 1, 1, 2018, 11, 12], dtype=object)
```

Ao fim do processo de conversão, as variáveis são dispostas em um novo *data frame*.

```
datafx_avaliacao = pd.DataFrame(X_avaliacao)
```

Em seguida, o *data frame* com as novas variáveis é exportado e novamente carregado.

```
datafx_avaliacao.to_csv(r"C:\Users\Spaox\Documents\Projeto\Arquivos\Tabelas\Exportadas\datafx_avaliacao_indicadores.csv", index = False)
datafx_avaliacao = pd.read_csv(r"C:\Users\Spaox\Documents\Projeto\Arquivos\Tabelas\Exportadas\datafx_avaliacao_indicadores.csv", index_col=0)
```

Assim é possível observar que as variáveis agora são numéricas.

```
datafx_avaliacao.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 90755 entries, 0 to 90754
Data columns (total 15 columns):
 #   Column   Non-Null Count  Dtype  
--- 
 0   0          90755 non-null   int64  
 1   1          90755 non-null   int64  
 2   2          90755 non-null   int64  
 3   3          90755 non-null   int64  
 4   4          90755 non-null   int64  
 5   5          90755 non-null   int64  
 6   6          90755 non-null   int64  
 7   7          90755 non-null   int64  
 8   8          90755 non-null   int64  
 9   9          90755 non-null   int64  
 10  10         90755 non-null   int64  
 11  11         90755 non-null   int64  
 12  12         90755 non-null   int64  
 13  13         90755 non-null   int64  
 14  14         90755 non-null   int64  
dtypes: int64(15)
memory usage: 10.4 MB
```

Para o processo de escalonamento dos valores é utilizada a ferramenta *Standard Scaler*. Assim, os valores dos atributos previsores foram padronizados.

```
from sklearn.preprocessing import StandardScaler

avaliacao_scaler = StandardScaler()
X_avaliacao_scaler = avaliacao_scaler.fit_transform(X_avaliacao)

X_avaliacao_scaler[0]
array([-0.12943797, -0.94874178,  0.56265362, -0.9193517 , -0.75717001,
       0.63177932,  0.33147337, -0.34344209,  0.42297415,  0.2036199 ,
      0.4564458 ,  0.0360818 ,  1.57330506,  1.33687182, -0.42094488])
```

Após o escalonamento dos valores, as variáveis dos atributos previsores e atributo classe foram separadas em bases de teste e treinamento através da biblioteca *train_test_split* do *Scikit Learn*.

Foram assim separados 80% dos registros para a base de treinamento e 20% para a base de teste. Através do parâmetro *random_state* é possível gerar a base de dados com os mesmos registros para comparar teste e treinamento.

```
X_avaliacao_treinamento, X_avaliacao_teste, y_avaliacao_treinamento, y_avaliacao_teste = train_test_split(X_avaliacao_scaler, y_avaliacao, test_size=0.2, random_state=42)

X_avaliacao_teste.shape, y_avaliacao_teste.shape
((18151, 15), (18151,))

X_avaliacao_treinamento.shape, y_avaliacao_treinamento.shape
((72604, 15), (72604,))
```

Após a divisão dos registros em bases de teste e treinamento, deu-se início ao desenvolvimento dos modelos e utilização dos algoritmos de *machine learning*.

O algoritmo de Árvore de Decisão (*Decision Tree*) foi o primeiro algoritmo a ser treinado. Primeiramente o algoritmo foi importado e iniciado o processo de ajuste das bases.

```
from sklearn.tree import DecisionTreeClassifier

dtc_avaliacao = DecisionTreeClassifier()
dtc_avaliacao.fit(X_avaliacao_treinamento, y_avaliacao_treinamento)

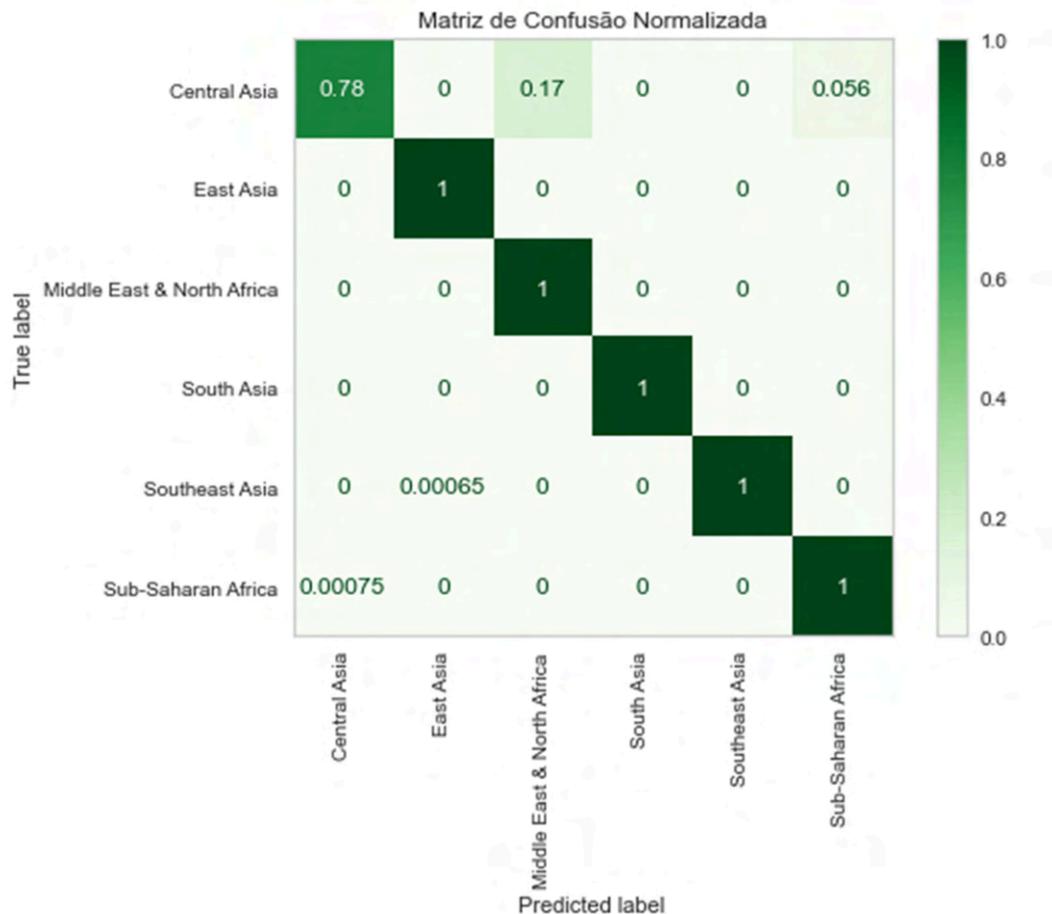
#previsões
dtc_avaliacao_yteste = dtc_avaliacao.predict(X_avaliacao_teste)      #previsões de teste
dtc_avaliacao_ytreinamento = dtc_avaliacao.predict(X_avaliacao_treinamento)#previsões de treinamento

print('Acurácia dos dados de Treinamento: {}'.format(accuracy_score(y_avaliacao_treinamento, dtc_avaliacao_ytreinamento)))
print('Acurácia dos dados de Teste: {}'.format(accuracy_score(y_avaliacao_teste, dtc_avaliacao_yteste)))

Acurácia dos dados de Treinamento: 0.9998760398870095
Acurácia dos dados de Teste: 0.9996143463170073
```

Ao obtermos os resultados da acurácia do algoritmo, pode-se observar o valor de 99% de acurácia nos dados de treinamento e teste.

Através da Matriz de Confusão é possível visualizar o desempenho de classificação do algoritmo.



Com o relatório de classificação é possível analisar medidas de avaliação do algoritmo.

```
print(classification_report(y_avaliacao_teste, dtc_avaliacao_yteste))

      precision    recall  f1-score   support

Central Asia       0.88     0.78     0.82      18
      East Asia      1.00     0.95     0.98      22
Middle East & North Africa  1.00     1.00     1.00    7410
      South Asia     1.00     1.00     1.00    6494
      Southeast Asia  1.00     1.00     1.00    1548
Sub-Saharan Africa 1.00     1.00     1.00   2659

accuracy           1.00
macro avg       0.98     0.96     0.97    18151
weighted avg     1.00     1.00     1.00    18151
```

O segundo algoritmo a ser testado foi o Random Forest. Após a importação da biblioteca foi realizado o processo de ajuste das bases.

```
from sklearn.ensemble import RandomForestClassifier

rfc_avaliacao = RandomForestClassifier()
rfc_avaliacao.fit(X_avaliacao_treinamento, y_avaliacao_treinamento)

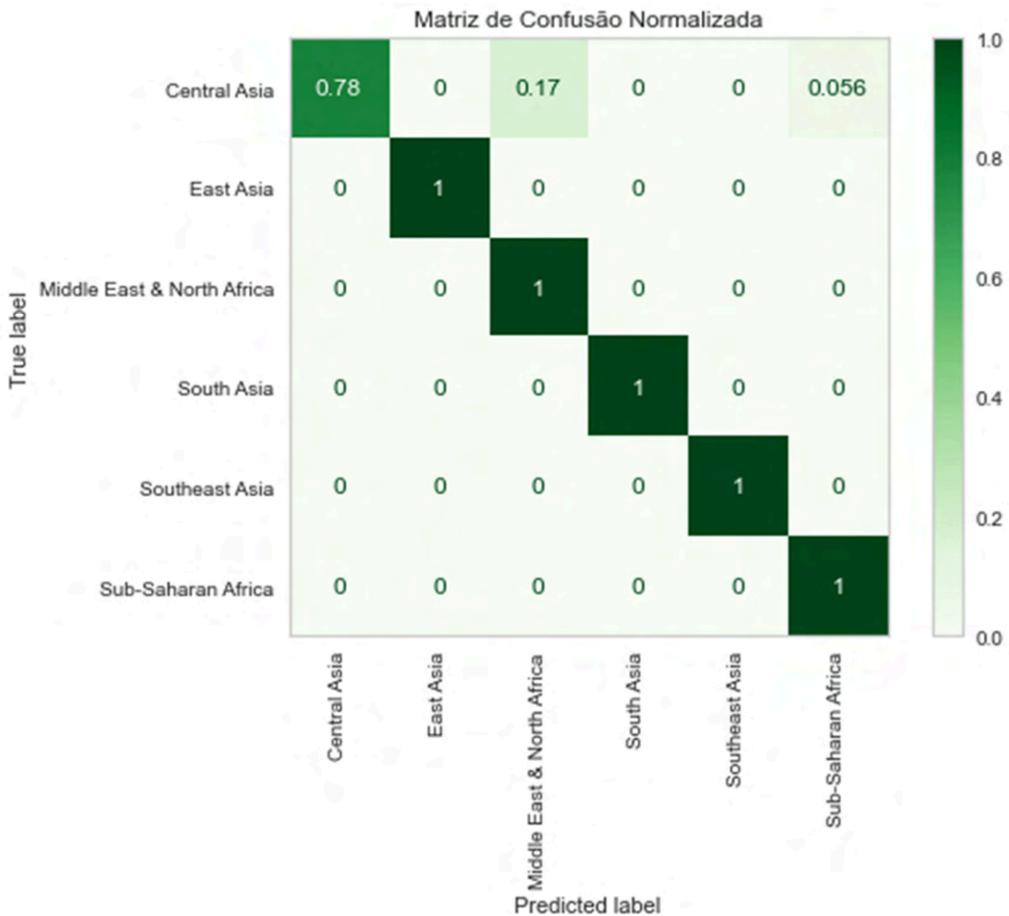
#previsões
rfc_avaliacao_yteste      = rfc_avaliacao.predict(X_avaliacao_teste)      #previsões de teste
rfc_avaliacao_ytreinamento = rfc_avaliacao.predict(X_avaliacao_treinamento)#previsões de treinamento

print('Acurácia dos dados de Treinamento: {}'.format(accuracy_score(y_avaliacao_treinamento, rfc_avaliacao_ytreinamento)))
print('Acurácia dos dados de Teste: {}'.format(accuracy_score(y_avaliacao_teste, rfc_avaliacao_yteste)))

Acurácia dos dados de Treinamento: 0.9998760398876095
Acurácia dos dados de Teste: 0.9997796264668614
```

É possível observar que os resultados de acurácia obtidos para os dados de treinamento e teste foram os semelhantes aos obtidos através do algoritmo de Árvore de Decisão, sendo 99% de acurácia para os dados de treinamento e teste.

Através da Matriz de Confusão é possível visualizar o desempenho de classificação do algoritmo.



Através do relatório de classificação é possível fazer uma análise sobre a capacidade de classificação do algoritmo.

```
print(classification_report(y_avaliacao_teste, rfc_avaliacao_yteste))

precision    recall  f1-score   support

Central Asia      1.00      0.78      0.88       18
East Asia         1.00      1.00      1.00       22
Middle East & North Africa  1.00      1.00      1.00     7410
South Asia        1.00      1.00      1.00     6494
Southeast Asia    1.00      1.00      1.00     1548
Sub-Saharan Africa 1.00      1.00      1.00     2659

accuracy                           1.00      18151
macro avg       1.00      0.96      0.98     18151
weighted avg     1.00      1.00      1.00     18151
```

Após o teste com o Random Forest, foi realizado um novo teste com o algoritmo KNN. Primeiramente foi feita a importação do algoritmo, assim como o processo de ajuste das bases.

```
from sklearn.neighbors import KNeighborsClassifier

knn_avaliacao = KNeighborsClassifier()
knn_avaliacao.fit(X_avaliacao_treinamento, y_avaliacao_treinamento)

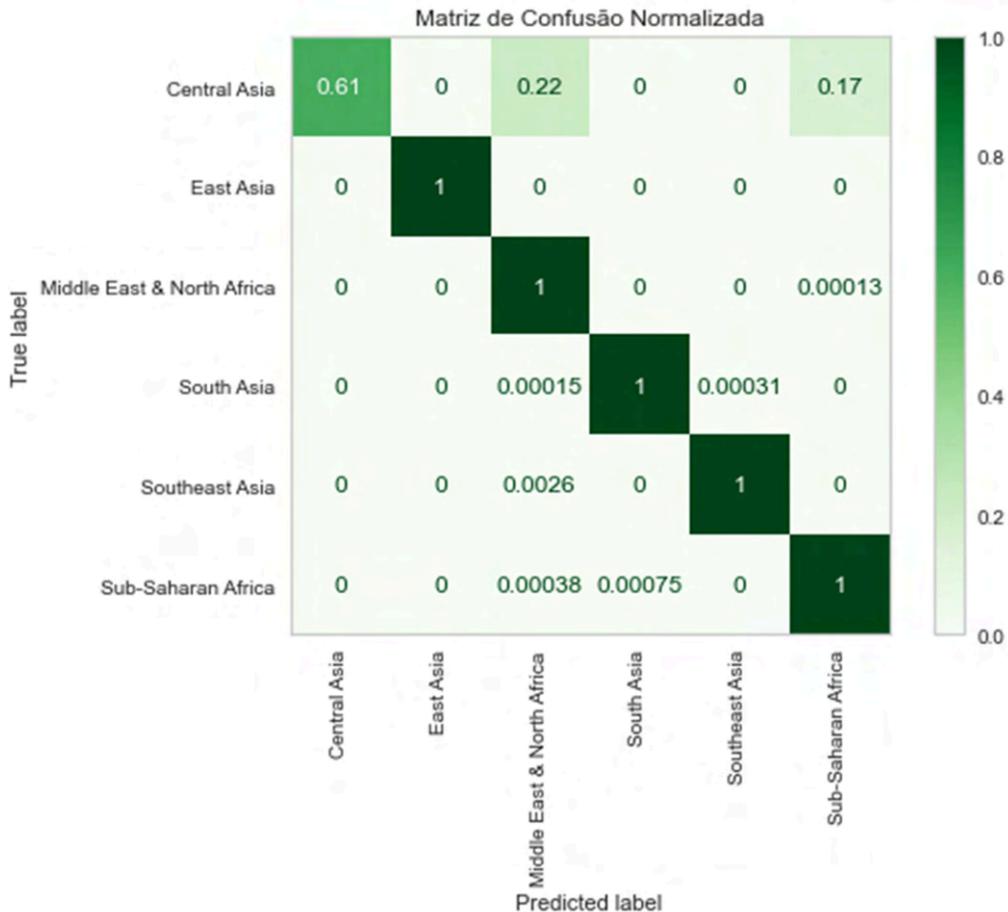
#previsoes
knn_avaliacao_yteste      = knn_avaliacao.predict(X_avaliacao_teste)      #previsores de teste
knn_avaliacao_ytreinamento = knn_avaliacao.predict(X_avaliacao_treinamento)#previsores de treinamento
```

```
print('Acurácia dos dados de Treinamento: {}'.format(accuracy_score(y_avaliacao_treinamento, knn_avaliacao_ytreinamento)))
print('Acurácia dos dados de Teste: {}'.format(accuracy_score(y_avaliacao_teste, knn_avaliacao_yteste)))|
```

Acurácia dos dados de Treinamento: 0.9991598259049088
Acurácia dos dados de Teste: 0.999008319100876

O algoritmo obteve também 99 % de acurácia para os dados de treinamento e teste.

Foram desenvolvidos modelos de visualização como a Matriz de Confusão e o relatório de classificação para análise de desempenho da classificação do algoritmo.



```
print(classification_report(y_avaliacao_teste, knn_avaliacao_yteste))
```

| | precision | recall | f1-score | support |
|----------------------------|-----------|--------|----------|---------|
| Central Asia | 1.00 | 0.61 | 0.76 | 18 |
| East Asia | 1.00 | 1.00 | 1.00 | 22 |
| Middle East & North Africa | 1.00 | 1.00 | 1.00 | 7410 |
| South Asia | 1.00 | 1.00 | 1.00 | 6494 |
| Southeast Asia | 1.00 | 1.00 | 1.00 | 1548 |
| Sub-Saharan Africa | 1.00 | 1.00 | 1.00 | 2659 |
| accuracy | | | 1.00 | 18151 |
| macro avg | 1.00 | 0.93 | 0.96 | 18151 |
| weighted avg | 1.00 | 1.00 | 1.00 | 18151 |

8. Apresentação dos resultados

8.1. Apresentação dos resultados da base de ataques terroristas

Para apresentação dos resultados foi elaborada uma tabela comparando as métricas de avaliação para todos os algoritmos.

Como mencionado anteriormente, a análise dos resultados será feita através da métrica F1-score da macro média.

| Algoritmo | Acurácia | Precisão | Sensitividade | F1-score |
|-------------------|----------|----------|---------------|----------|
| Árvore de Decisão | 0.98 | 0.87 | 0.87 | 0.87 |
| Random Forest | 0.97 | 0.81 | 0.66 | 0.69 |
| KNN | 0.79 | 0.76 | 0.51 | 0.53 |

Ao analisar os resultados de acurácia obtidos pelos algoritmos, pode-se observar que os três possuíram resultados muito positivos, com destaque para o algoritmo de Árvore de Decisão com 98% de acurácia e para o algoritmo Random Forest que obteve 97% de acurácia. Porém, se tratando de uma base desbalanceada, pode não ser a melhor métrica de avaliação para os algoritmos, para isso, serão analisadas outras métricas, como a precisão, sensitividade e o F1-score, além da matriz de confusão.

Observando os valores obtidos através da visualização da matriz de confusão de todos os algoritmos testados pode-se destacar o algoritmo de Árvore de Decisão. Além de obter valores próximos a 1.00 em grande parte da diagonal principal da matriz de confusão, foi o algoritmo que mais classificou corretamente as classes “Central Asia” e “East Asia” que possuem o menor número de registros testados e treinados entre todas as classes.

O algoritmo que obteve o melhor resultado prevendo os reais positivos verdadeiros foi o algoritmo de Árvore de Decisão. Em uma comparação geral entre os três algoritmos, o algoritmo de Árvore de Decisão obteve um resultado de 0.87 de precisão, enquanto os algoritmos Random Forest e KNN obtiveram resultados de 0.81 e 0.76, respectivamente. Explorando os valores de precisão para cada classe, é possível observar que o algoritmo de Árvore de Decisão obteve valores altos na classificação de todas as regiões (ou classes), com maior destaque para as regiões “Middle East & North Africa” e “Southeast Asia”, ambas com valores de 0.99. Entre os valores obtidos para as classes Central Asia, Middle East & North Africa, South Asia, Southeast Asia e Sub-Saharan Africa o algoritmo de Árvore de Decisão obteve os melhores resultados, porém os algoritmos Random Forest e KNN obtiveram melhores resultados de precisão para a classe “East Asia”, onde ambos tiveram valores de 1.00.

Entre os resultados de sensitividade (recall), o algoritmo de Árvore de Decisão obteve a maior macro média em relação aos outros dois algoritmos, isto é, o resultado obtido pelo algoritmo para a classificação correta das classes foi de 0.87. Analisando a métrica de avaliação de sensitividade entre as classes pode-se observar que em todas o algoritmo de Árvore de Decisão obteve os melhores valores entre todos os algoritmos testados.

Da mesma maneira ao analisar as outras métricas de avaliação é possível observar através da tabela com os valores obtidos no relatório de classificação que o algoritmo de Árvore de Decisão obteve o melhor resultado para a métrica F1-score, com o valor de 0.87. Os outros algoritmos obtiveram resultados abaixo: o algoritmo Random Forest obteve uma média de 0.69, enquanto o algoritmo KNN obteve uma média de 0.53. Entre todas as classes, o algoritmo de Árvore de Decisão também obteve melhores resultados, e mesmo nas classes onde os resultados foram menores, os valores ainda assim foram acima de 0.60. Explorando os resultados entre as classes, é possível observar uma maior variação no algoritmo KNN, enquanto os resultados do algoritmo Random Forest mostram valores acima de 0.90 para as classes com mais registros e valores bem menores para classes com menos registros.

Diferentemente da média ponderada, o F1-score da macro média calcula a média harmônica entre as classes. Essa métrica foi utilizada como determinante para avaliação dos algoritmos devido ao desbalanceamento das classes.

Ao analisar todas as métricas é possível perceber que o algoritmo de Árvore de Decisão obteve melhores resultados em todas as medidas de avaliação, inclusive na métrica utilizada para determinar o desempenho do algoritmo para as classificações.

Dessa forma é possível observar que os resultados obtidos refletem em sua maioria a frequência com que eventos terroristas acontecem nas regiões do continente asiático que mais sofrem com atos dessa natureza. Já nas regiões que possuem menos registros de eventos terroristas há maiores dificuldades de se prever novos ataques. Assim, através do estudo das informações sobre ataques terroristas é possível realizar com precisão o mapeamento de áreas mais suscetíveis a eventos dessa natureza.

8.2. Apresentação dos resultados da base de indicadores políticos e sociais em conjunto com registros de ataques terroristas

Semelhante a análise feita para os registros e informações de ataques terroristas, foi elaborada uma tabela comparativa entre os resultados dos algoritmos com dados de indicadores políticos, sociais e culturais.

| Algoritmo | Acurácia | Precisão | Sensitividade | F1-score |
|-------------------|----------|----------|---------------|----------|
| Árvore de Decisão | 0.99 | 0.98 | 0.96 | 0.97 |
| Random Forest | 0.99 | 1.00 | 0.96 | 0.98 |
| KNN | 0.99 | 1.00 | 0.93 | 0.96 |

Inicialmente, pode-se perceber que os valores obtidos para cada métrica foram muito altos, próximos a 1.00. Os resultados obtidos podem ser explicados pela falta de complexidade das informações contidas no *dataset*.

Primeiramente, pode-se observar que todos os algoritmos obtiveram o mesmo valor de 0.99 de acurácia. Os resultados podem ser observados através da matriz de confusão desenvolvida para cada algoritmo. Entre todas as classes, apenas a classe “Central Asia” não obteve o valor de 1.00 na diagonal principal. Enquanto o algoritmo Random Forest apresentou valores exatos de classificação para grande parte das classes, houve uma pequena variação para os algoritmos de Árvore de Decisão e KNN.

Entre os três algoritmos testados é possível observar que tanto o algoritmo Random Forest quanto o KNN obtiveram 1.00 de precisão, isto é, todos os registros verdadeiramente positivos foram classificados de maneira correta. Já os valores de precisão obtidos pelo algoritmo de Árvore de Decisão foram semelhantes aos outros dois algoritmos, exceto para a classe “Central Asia”, onde houve um valor mais abaixo em comparação aos outros dois algoritmos, dessa forma a macro média obtida pelo algoritmo de Árvore de Decisão foi de 0.98.

Entre os resultados de sensitividade (recall), os algoritmos de Árvore de Decisão e Random Forest obtiveram os maiores resultados da macro média, onde ambos os resultados obtidos pelos algoritmos para a classificação correta das classes foi de 0.96, enquanto o algoritmo KNN obteve 0.93 nos resultado da métrica. Analisando a métrica de avaliação de sensitividade entre as classes pode-se observar que todos os algoritmos identificaram falsos negativos na classe “Central Asia”, com um número menor apresentado pelo algoritmo KNN, o que foi determinante para que o algoritmo apresentasse menor resultado nessa métrica.

Analizando a métrica de avaliação F1-score, onde obtém-se a média harmônica dos resultados das classes, pode-se observar que o algoritmo Random Forest obtém o melhor resultado, já que o algoritmo obteve melhores resultados nas métricas de precisão e sensitividade.

Após a análise das métricas de avaliação é possível perceber que a utilização de um *dataset* com valores de pouca complexidade podem gerar resultados melhores, mas com baixa capacidade de generalização. Dessa forma, analisando as macro médias obtidas após o teste dos algoritmos pode-se observar que combinando registros de ataques terroristas a informações superficiais sobre política, sociedade e cultura em regiões onde há maiores ocorrências de ataques terroristas, é possível determinar padrões em áreas mais propensas a eventos dessa natureza.

9. Conclusão

Após a análise dos dados e informações obtidos nesta pesquisa é possível concluir que qualquer sociedade no mundo está propensa a enfrentar eventos como ataques terroristas. Embora exemplos mais conhecidos como o ataque ao World Trade Center em 2001 sejam considerados referências para o que hoje pensamos sobre ataques terroristas, pode-se concluir que mesmo eventos de menor escala, que acontecem em regiões mais distantes e que atingem um número menor de pessoas também podem ser classificados dessa maneira.

Através desse estudo pode-se perceber que as regiões do continente asiático e africano registraram inúmeras ocorrências de ataques terroristas em uma década onde crises humanitárias se agravaram em ambos os continentes, guerras civis despontaram e o número de descolados aumentou consideravelmente, intensificando o fluxo migratório. Os reflexos de crises e conflitos podem ser sentidos nas sociedades, já que através da pesquisa pode-se observar que territórios que enfrentaram guerras civis durante esse século como o Iraque, Afeganistão e Somália registram altos números de deslocados e ataques terroristas no período estudado.

A diversidade cultural, étnica e religiosa presente em qualquer sociedade pode ser transformada em estímulo para conflitos dessa natureza, e mesmo a população está vulnerável a eventos assim, mesmo que o objetivo seja causar um impacto direto ao Estado. Conflitos devido a diferenças políticas e sociais, a reivindicação de territórios de forma radical e perseguições a minorias são problemas presentes desde

o início da humanidade, mas que nos dias de hoje podem causar ainda mais impactos, tanto pelos estragos que a evolução bélica pode causar, quanto pela forma como esse tipo de informação é recebida atualmente, podendo gerar novos eventos.

Atualmente não há qualquer solução que não seja considerada utópica em face de tantos registros reunidos ao longo dos anos sobre ataques terroristas e conflitos de escalas maiores. Mesmo que hoje hajam estudos a fim de entender do que é feito o terrorismo, ainda há uma grande distância a ser percorrida para encontrar medidas de previsão e prevenção de ataques.

10. Links

Link para o vídeo: <https://www.youtube.com/watch?v=wLpfOOFJzyY>

| | | | | |
|---|-------------|----------|--------------------|----------------|
| Link | para | o | repositório | Github: |
| https://github.com/Charlesep1996/TCCPUCMG.git | | | | |

Referências

Internal Displacement Monitoring Centre – IDMC. Global Report on Internal Displacement, 2021. Disponível em: <https://www.ohchr.org/en/migration>

Concern Worldwide U.S. The 10 largest refugee crises to follow in 2022, 2022. Disponível em: <https://www.concernusa.org/story/largest-refugee-crises/>

Organização das Nações Unidas – ONU. Refugee Data Finder, 2022. Disponível em: <https://www.unhcr.org/refugee-statistics/>

Organização das Nações Unidas – ONU. Doadores mostram solidariedade com os refugiados e garantem apoio aos programas do ACNUR em 2021, 2021. Disponível em: <https://brasil.un.org/pt-br/106900-doadores-mostram-solidariedade-com-os-refugiados-e-garantem-apoio-aos-programas-do-acnur-em>

Portal da Imigração - Ministério da Justiça e Segurança Pública. Relatório Refúgio em Números, 2020. Disponível em: <https://portaldeimigracao.mj.gov.br/pt/dados/refugio-em-numeros>

International Rescue Committee – IRC. Migrants, asylum seekers, refugees and immigrants: What's the difference? 2022. Disponível em: <https://www.rescue.org/article/migrants-asylum-seekers-refugees-and-immigrants-whats-difference>

BBC News Brasil. Os 10 países que concentram 75% dos ataques terroristas no mundo, 2017. Disponível em: <https://www.bbc.com/portuguese/internacional-40655023>

Federal Bureau of Investigation – FBI. Terrorism. Disponível em: <https://www.fbi.gov/investigate/terrorism>
GSDRC. Causes of Terrorism: An Expanded and Updated Review of the Literature, 2004. Disponível: <https://gsdrc.org/document-library/causes-of-terrorism-an-expanded-and-updated-review-of-the-literature/>

Proceedings of the National Academy of Sciences - Pnas. Armed-conflict risks enhanced by climate-related disasters in ethnically fractionized countries, 2016. Disponível em: <https://www.pnas.org/doi/10.1073/pnas.1601611113>

Portal da Imigração - Ministério da Justiça e Segurança Pública. Relatório Refúgio em Números, 2019. Disponível em: <https://portaldeimigracao.mj.gov.br/pt/dados/refugio-em-numeros>

Olusola A. Olabanjo, Benjamin S. Aribisala, Manuel Mazzara, Ashiribo S. Wusu. An ensemble machine learning model for the prediction of danger zones: Towards a global counter-terrorism, Soft Computing Letters, Volume 3, 2021. Disponível em: <https://www.sciencedirect.com/science/article/pii/S2666222121000101>

Musumba M, Fatema N, Kibriya S. Prevention Is Better Than Cure: Machine Learning Approach to Conflict Prediction in Sub-Saharan Africa. *Sustainability*, 2021; 13(13):7366. Disponível em: <https://doi.org/10.3390/su13137366>

Nurunnabi, Mohammad & Sghaier, Asma. Socioeconomic Determinants of Terrorism. *Digest of Middle East Studies*, 2018; 27. 10.1111/dome.12139.

Geeks for Geeks. Decision Tree. Disponível em:
<https://www.geeksforgeeks.org/decision-tree/>

Xoriant. Decision Tree for Classification: A Machine Learning Algorithm. Disponível em: <https://www.xoriant.com/blog/decision-trees-for-classification-a-machine-learning-algorithm>

EDUCBA. Decision Tree Advantages and Disadvantages. Disponível em:
<https://www.educba.com/decision-tree-advantages-and-disadvantages/>

Jigsaw Academy. Decision Tree in Machine Learning: Types, Advantages, Disadvantages in 5 Points. Disponível em:
<https://www.jigsawacademy.com/blogs/data-science/decision-tree-in-machine-learning/>

IBM. Random Forest. Disponível em: <https://www.ibm.com/cloud/learn/random-forest>

Adobe Experience League. Algoritmo Random Forest. Disponível em:
<https://experienceleague.adobe.com/docs/target/using/activities/automated-personalization/algo-random-forest.html?lang=pt-BR>

Sruthi E R. Understanding Random Forest, 2022. Disponível em:
<https://www.analyticsvidhya.com/blog/2021/06/understanding-random-forest/>

Java T Point. K-Nearest Neighbor (KNN) Algorithm for Machine Learning. Disponível em: <https://www.javatpoint.com/k-nearest-neighbor-algorithm-for-machine-learning>

Links datasets

Educational Index

https://hdr.undp.org/*

<https://hdr.undp.org/en/indicators/103706> (Link indisponível)

Flags of the World (Dashboard Power BI).

<https://www.worldometers.info/geography/flags-of-the-world/>

Global Terrorism Database.

<https://www.start.umd.edu/gtd/>

Human Development Index (HDI)

https://hdr.undp.org/*

Historical Index of Ethnic Fractionalisation Dataset

<https://dataverse.harvard.edu/dataset.xhtml?persistentId=doi:10.7910/DVN/4JQRCL>

Life expectancy at birth (years)

https://hdr.undp.org/*

<https://hdr.undp.org/en/indicators/69206> (Link indisponível)

List of all countries with their 2 digit codes (ISO 3166-1)

<https://datahub.io/core/country-list#data-cli>

Political Regime

https://github.com/owid/notebooks/tree/main/BastianHerré*

https://github.com/owid/notebooks/tree/main/BastianHerré/political_regimes (Link indisponível)

Refugee Data Finder

<https://www.unhcr.org/refugee-statistics/download/?url=CYS8We>

Religious Composition by Country, 2010-2050

<https://www.pewresearch.org/religion/2015/04/02/religious-projection-table/>

Unemployment, total

https://hdr.undp.org/*

<https://hdr.undp.org/en/indicators/140606> (Link indisponível)

Worldwide Governance Indicator

<http://info.worldbank.org/governance/wgi/>

*Alguns links originais dos datasets foram alterados após o descarregamento dos dados e do desenvolvimento da pesquisa. Optou-se por adicionar à relação os links principais dos órgãos e pesquisadores que disponibilizaram as informações.