# DESAFIO CIVITAS – EMD

# Jorge Marcelino dos Santos Júnior

**Resumo:** Neste trabalho apresentamos os resultados obtidos da análise de leituras individuais de radares espalhados pelos diversos bairros da cidade do Rio de Janeiro. Utilizando os dados de uma tabela BigQuery, fornecida pela Central de Inteligência, Vigilância e Tecnologia de Apoio à Segurança Pública (CIVITAS), foi possível, através da análise exploratória dos dados e eliminando suas inconsistências, desenvolver um script simples que possibilitou a identificação de possíveis casos de clonagem de placas de veículos.

**Introdução**

A clonagem de placas de veículos é um problema crescente que afeta a cidade do Rio de Janeiro, causando preocupações significativas tanto para as autoridades de segurança quanto para os cidadãos. Este crime consiste na duplicação de placas de automóveis legítimos e sua utilização em veículos não autorizados, muitas vezes para perpetrar outros crimes ou escapar de penalidades legais.

A clonagem de placas no Rio de Janeiro tem várias ramificações. Criminosos utilizam veículos clonados para cometer delitos como roubos, sequestros e tráficos de drogas, sabendo que, ao serem capturados em câmeras de vigilância, a responsabilidade recairá sobre o proprietário do veículo original, cuja placa foi clonada. Isso não apenas dificulta a aplicação da lei, mas também resulta em transtornos e prejuízos para os cidadãos inocentes que têm suas placas clonadas.

A CIVITAS é um projeto tecnológico avançado de monitoramento, concebido para implementar um cerco inteligente e fortalecer a segurança pública municipal. Este projeto capitaliza a vasta experiência e conhecimento técnico do Centro de Operações Rio (COR) e se beneficia da tradição e da credibilidade do Disque Denúncia, um sistema de atendimento com quase três décadas de existência, reconhecido como uma referência para os cidadãos do estado do Rio de Janeiro. O Disque Denúncia, como órgão de interação com a população poderá, dentro do seu escopo de trabalho, integrar o CIVITAS como uma rede de vigilância comunitária, a partir de seu funcionamento 24/7 ou seja, 24 horas por dia, 7 dias por semana.

O sistema CIVITAS utiliza imagens de câmeras do COR e da CET-Rio e planeja, no futuro, incorporar imagens de empresas de segurança privada. Para o monitoramento de veículos, o projeto conta com 900 radares e 50 câmeras equipadas com tecnologia de leitura de placas. Esses dispositivos são fundamentais na identificação de veículos clonados, aprimorando a capacidade de resposta e intervenção na cidade.

**Amostra**

A amostra utilizada neste trabalho está disponíveis na tabela BigQuery rj-cetrio.desafio.readings\_2024\_06, de uso restrito da CIVITAS. Abaixo apresentamos o detalhamento das colunas que compõem a tabela de dados.

Tabela1 – A primeira coluna apresenta o nome de cada uma das colunas da tabela Big Query utilizada. A segunda coluna descreve o tipo dos dados alocados em cada coluna da tabela de dados e a terceira coluna da tabela abaixo, apresenta uma descrição sucinta das colunas da tabela de dados Bigquery.

|  |  |  |
| --- | --- | --- |
| **Coluna** | **Tipo** | **Descrição** |
| datahora | TIMESTAMP | Data e hora da leitura |
| datahora\_captura | TIMESTAMP | Data e hora da captura pelo radar |
| placa | BYTES | Placa do veículo capturado |
| empresa | STRING | Empresa do veículo |
| tipoveiculo | STRING | Tipo do veículo |
| velocidade | INTEGER | Velocidade do veículo |
| camera\_numero | STRING | Número identificador da câmera |
| camera\_latitude | FLOAT | Latitude da câmera |
| camera\_longitude | FLOAT | Longitude da câmera |

A tabela inicial foi construída por informações obtidas entre os dia 03/06/2024, às 00h00min00s UTC e o dia10/06/2024, às 14h31min56s UTC (cerca de uma semana), com as colunas discriminadas na Tabela 1.

Uma informação importante é que as placas foram substituídas por códigos para impossibilitar a sua identificação. Apenas o programa que gerou os códigos pode desencriptar o mesmo.

**Metodologia**

A primeira abordagem foi a realização de uma análise visual da tabela para a localização de possíveis inconsistências que pudesse afetar o desenvolvimento da QuerySQL, e por consequentemente, o objetivo do trabalho: a identificação de possíveis placas clonadas. Isto tudo baseado em registros de leitura de velocidade e localização de câmeras, além das informações de data e horário dos eventos. O código completo está apresentado no Anexo deste trabalho.

Após a análise visual, iniciamos com a exclusão das linhas de valor *null*, ou seja, sem informações. Além disso, eliminamos de imediato as linhas de velocidade igual a zero, por entender que os radares apresentaram problemas, uma vez que o carro, ao passar pelo radar não poderia apresentar velocidade zero. Por desconhecer o modelo do radar ficamos impossibilitado de conhecer o erro da medida.

As linhas que apresentavam alguma das coordenadas geográficas de latitude ou longitude igual a zero também foram excluídas por ser um erro da leitura do GPS ou do funcionamento geral do radar, pois a cidade do Rio de Janeiro está localizada no Hemisfério Sul, com coordenadas oscilando próximo a -23 graus de latitude e -43,3 graus de longitude, distante do ponto de origem das coordenadas geográficas (lat = 0o, lon = 0o), no Oceano Atlântico.

O próximo passo foi a identificação de leitura de placas duplicadas, ou seja, verificar quais as placas que apresentavam mais de uma entrada na tabela. Esta nova amostra passou a ser a fonte de interesse do trabalho, pois a leitura única impossibilita a análise de suspeita de clonagem como apresentada nesta abordagem.

Como temos apenas placas que apresentavam mais de uma passagem pelos radares, calculamos a diferença de tempo entre estas passagens dos veículos em cada um dos radares utilizando a função TIMESTAMP\_DIFF.

Com as funções ST\_DISTANCE e ST\_GEOPOINT, aplicadas às coordenadas geográficas dos radares, foi possível determinar a distância linear entre os radares (*ΔS*).

Todos estes parâmetros foram selecionados para podermos utilizar na aproximação da seleção de placas suspeitas de clonagem. O termo aproximação é realmente o mais adequado porque, na hipótese utilizada, calculando o intervalo de tempo entre a passagem da placa por dois radares diferentes e calculando o tempo para percorrer a distância linear entre os mesmos dois radares, podemos determinar se é possível a ocorrência de tal fato.

Assim, , onde *dtcalc* é o intervalo de tempo mínimo necessário, dentro dos dados, para percorrer a distância calculada pelas coordenadas geográficas com a maior velocidade observada pelos radares (*ΔV*).

O parâmetro *dtint* é o intervalo de tempo entre as observações, das mesmas placas em locais distintos, utilizado o *timestamp* fornecido pela tabela.

A hipótese assumida é a de que os veículos que apresentam as placas clonadas foram observados em um intervalo de tempo inferior ao tempo necessário para se deslocar de um ponto de observação (radar1) a outro (radar2).

Avaliar através desta hipótese pode levar a conclusões muito próximas da realidade, porém não leva em conta as diversas características geográficas e demográficas da cidade. Melhorias no processo serão descritas na seção Propostas de Melhorias.

**Resultados**

Após todas as análises exploratórias dos dados e a eliminação das inconsistências encontradas na tabela, que compreende aproximadamente o período de uma semana em junho de 2024, foi desenvolvido uma Query SQL no intuito de selecionar placas de veículos com a possibilidade de terem sidos clonadas.

Neste trabalho foram encontradas duas placas de veículos que satisfizeram a hipótese de intervalo de tempo de observação, em dois diferentes radares, inferior ao intervalo mínimo de deslocamento entre os estes dois mesmos radares.

Como exemplo, podemos citar o caso de um veículo, que em um intervalo de tempo de 1 segundo foi observado por dois radares distantes 1,25 km. No segundo caso, a distância entre os radares é superior a 4 km e o veículo foi observado com um intervalo de 23 segundos entre os eventos.

Como dito anteriormente, as placas foram encriptadas para impossibilitar a identificação. Porém conseguimos determinar os locais de instalação dos radares por onde passaram os carros com suspeita de clonagem. A primeira placa foi encontrada em veículos no bairro de Irajá e a segunda placa estava em veículos nos bairros da Taquara e Jacarepaguá.

**Propostas de melhorias**

Como apresentado anteriormente, a cidade do Rio de Janeiro apresenta diversas características geográficas e demográficas que podem influenciar acentuadamente na determinação das distâncias reais percorridas entre os diferentes radares distribuídos em nossa gigantesca malha rodoviária.

Uma alternativa, ou aprimoramento desta abordagem, seria a utilização da API do Google Maps para a determinação da distância entre os radares utilizando as vias que possam minimizar todo o trajeto. Assim teríamos a distância por meio de vias públicas reais e não apenas a distância linear.

Uma segunda alternativa ou complementando a primeira, seria verificar se o aplicativo WAZE, que já tem parceria com a prefeitura para alertas, possui algum tipo de memória do trânsito da cidade que possa ser utilizada durante a análise de deslocamento e de tempo de trajeto, de modo a deixar a verificação das placas o mais acurado possível. Assim poderia ser utilizado o tempo de deslocamento levando em conta o percurso mais curto e a velocidade média do deslocamento no horário da captura das imagens.

Estas sugestões possuem uma ação que necessita ser avaliada, o tempo de processamento e memória necessária.

**Conclusão**

Ao término deste trabalho encontramos que:

1. Os dados fornecidos pelos radares permitem a realização de uma busca de veículos suspeitos de estarem utilizando placas clonadas.
2. As aproximações necessárias para o desenvolvimento da Query SQL mostraram-se coerentes, em observância com a realidade, permitindo reconhecer dois casos suspeitos.
3. A clonagem de placas de veículos no Rio de Janeiro representa um problema complexo que requer uma abordagem multifacetada. A combinação de tecnologia avançada, análise de dados, colaboração comunitária e parcerias estratégicas forma a base de uma estratégia eficaz para combater este e outros tipos de delitos. Somente através de esforços coordenados e contínuos será possível reduzir a incidência de crimes e melhorar a segurança pública na cidade.

Referências

# Prefeitura lança Central de Inteligência, Vigilância e Tecnologia de Apoio à Segurança Pública (CIVITAS) e anuncia parceria com Disque Denúncia,

<https://prefeitura.rio/casa-civil/prefeitura-lanca-central-de-inteligencia-vigilancia-e-tecnologia-de-apoio-a-seguranca-publica-civitas-e-anuncia-parceria-com-disque-denuncia/> , disponível em 04/06/2024, acesso em 11/06/2024.

# Rio inaugura Central de Inteligência para auxiliar forças de segurança; radares serão usados em cercos eletrônicos

<https://g1.globo.com/rj/rio-de-janeiro/noticia/2024/06/04/central-de-informacoes-e-inaugurada-em-parceria-entre-a-prefeitura-do-rio-e-disque-denuncia.ghtml>, disponível em 04/06/2024, acesso em 11/06/2024.

**ANEXO**

CÓDIGO COMENTADO

-- Selecionando os registros não nulos e válidos

CREATE TEMPORARY TABLE ValidRecords AS

SELECT datahora, datahora\_captura, placa, velocidade, camera\_latitude, camera\_longitude

FROM rj-cetrio.desafio.readings\_2024\_06

WHERE datahora IS NOT NULL

AND datahora\_captura IS NOT NULL

AND placa IS NOT NULL

AND empresa IS NOT NULL

AND tipoveiculo IS NOT NULL

AND velocidade IS NOT NULL

AND camera\_numero IS NOT NULL

AND camera\_latitude IS NOT NULL

AND camera\_longitude IS NOT NULL

-- eliminando os erros de leituras de velocidade dos radares = 0

AND velocidade != 0.0

-- eliminando os erros de posicionamento dos radares

AND camera\_latitude != 0.0

AND camera\_longitude != 0.0;

-- Identificando leituras de placas duplicadas

CREATE TEMPORARY TABLE Duplicates AS

SELECT \*

FROM ValidRecords

GROUP BY datahora, datahora\_captura, placa, velocidade, camera\_latitude, camera\_longitude

HAVING COUNT(placa) > 1;

-- Calculando a distância entre os pontos de observação

CREATE TEMPORARY TABLE DistanceCalculation AS

SELECT

-- calculando a diferença de tempo entre duas obserções de placas (em segundos)

t1.placa,

t1.datahora AS datahora1,

t2.datahora AS datahora2,

timestamp\_diff(t2.datahora, t1.datahora, second) AS dt\_int,

-- calculando a distância entre os radares que capturaram as placas

t1.camera\_latitude AS lat1,

t1.camera\_longitude AS lon1,

t2.camera\_latitude AS lat2,

t2.camera\_longitude AS lon2,

ST\_DISTANCE(ST\_GEOGPOINT(t1.camera\_longitude, t1.camera\_latitude), ST\_GEOGPOINT(t2.camera\_longitude, t2.camera\_latitude))/1000 AS distancia,

-- utilizando a maior velocidade observada como parâmetro para calcular o tempo de deslocamento entre os radares

GREATEST(t1.velocidade, t2.velocidade) AS max\_velocidade

FROM

Duplicates t1

-- fazendo o join para selecionar apenas as placas duplicadas

JOIN

Duplicates t2

ON

t1.placa = t2.placa

WHERE

t1.datahora < t2.datahora;

-- Calculando dt\_calc (intervalo de tempo calculado entre os radares na maior velocidade registrada)

CREATE TEMPORARY TABLE FinalCalculation AS

SELECT

\*,

distancia / max\_velocidade AS dt\_calc

FROM

DistanceCalculation;

-- Filtrando as placas onde dt\_int < dt\_calc (o intervalo de tempo entre as observações é menor que o intervalo calculado)

SELECT

placa, datahora1, datahora2, dt\_int, distancia, max\_velocidade, dt\_calc, lat1, lon1, lat2, lon2

FROM

FinalCalculation

WHERE

(dt\_int/360) < dt\_calc

AND distancia != 0;

-- Limpando as tabelas temporárias

DROP TABLE IF EXISTS ValidRecords;

DROP TABLE IF EXISTS Duplicates;

DROP TABLE IF EXISTS DistanceCalculation;

DROP TABLE IF EXISTS FinalCalculation;