Processo Seletivo

Time: Insights
Vaga: Engenheiro de Machine Learning

Desafio Técnico

Objetivo principal - Produtização de um modelo de regressão

Case

A rede de farmácias Campifarma é um tradicional varejista que atua exclusivamente no município de Campinas. Atualmente, eles possuem 289 unidades, como pode ser visualizado no mapa abaixo. No entanto, nos últimos anos eles vêm sistematicamente perdendo mercado, principalmente para grandes redes com presença nacional, como as redes “Pague Menos”, “Drogasil”, “Drogaria São Paulo”, “Onofre” e outras.

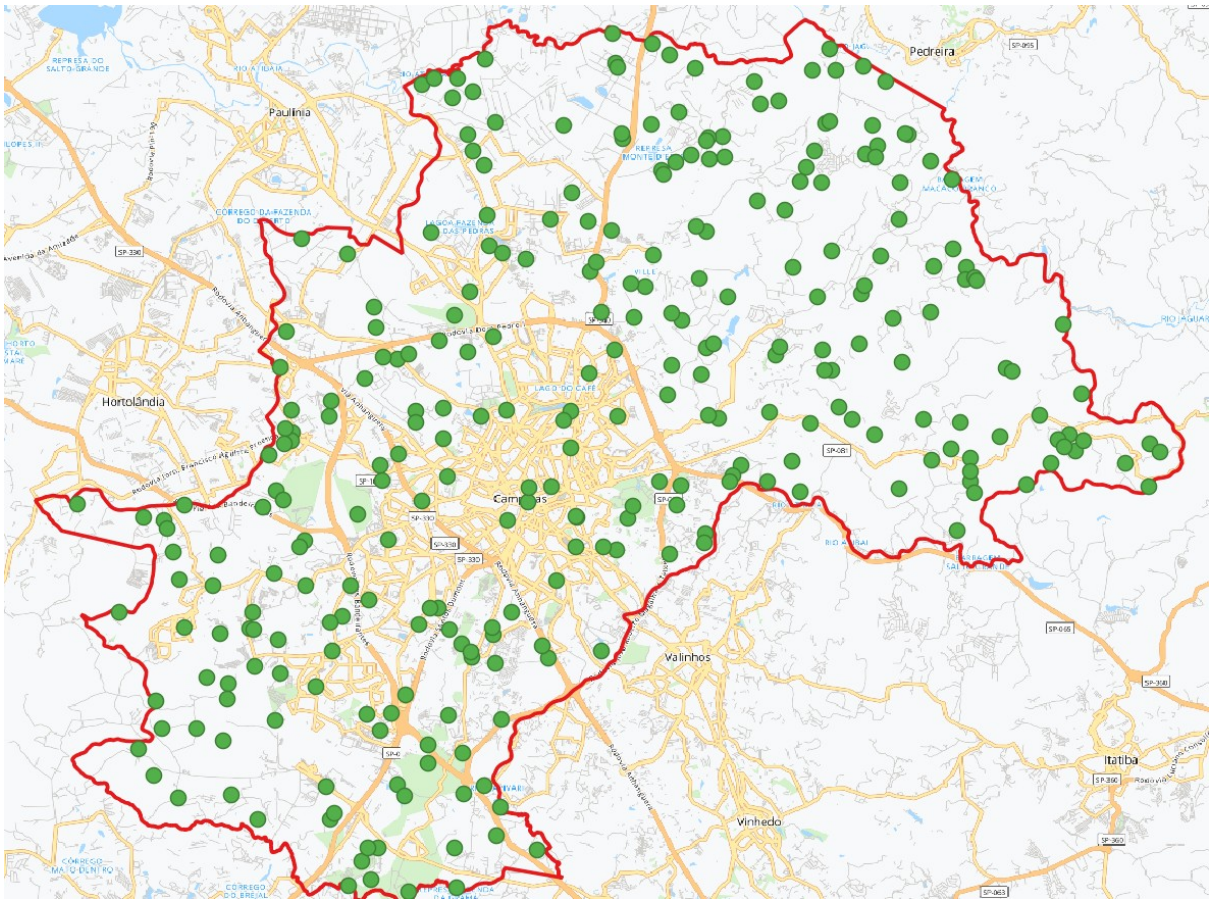

Figura 1 - Visualização das unidades da rede Campifarma no município de Campinas.

Assim, eles contrataram a Geofusion para ajudá-los a enfrentar este momento difícil da empresa. Nossa equipe de Consultoria, após diversas

análises dos dados de venda da rede Campifarma, chegou a conclusão que a melhor estratégia para a recuperação do seu *share* do mercado seria a conversão de algumas pequenas farmácias, além da abertura de novas lojas no município de Campinas.

Em seguida, nossa equipe de Cientistas de Dados de posse do faturamento médio mensal de cada unidade da rede Campifarma identificou que a quantidade de alguns pontos de interesse (POIs - *Points of Interest*) na proximidade da mesma, como escolas, lotéricas, agências bancárias e outras, eram determinantes para explicar o faturamento de cada uma das unidades da rede Campifarma. Após diversos testes, os Cientistas de Dados chegaram a conclusão que uma área de influência de 1000 metros, centrada da unidade, era ideal para explicar o faturamento dela. À esta área de influência por raio, dá-se o nome de isocota.

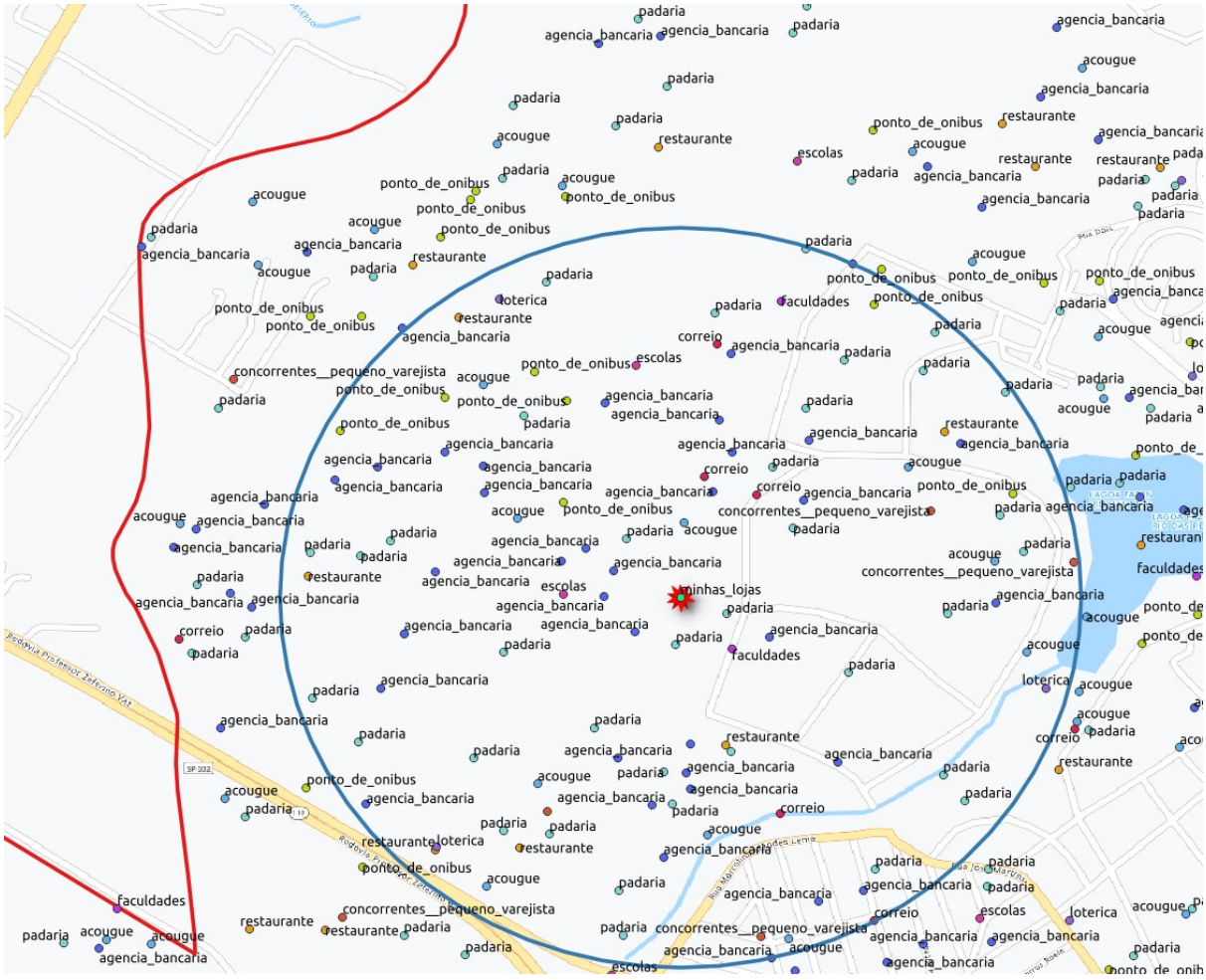

Figura 2 - Exemplo de uma unidade de rede Campifarma que contém diversos POIs em sua área de influência (ou isocota) de 1000 metros.

Assim, foi criado um modelo de regressão, em um jupyter-notebook, que dado um ponto e seu entorno prevê o faturamento de uma eventual unidade da rede Campifarma neste ponto. Após a sua elaboração, o modelo foi utilizado para se mensurar o retorno obtido ao se abrir uma loja em um novo ponto ou na conversão de uma farmácia já existente neste ponto, caso, naturalmente, já exista uma farmácia na proximidade dele.

Na entrega do projeto, após validações com os analistas de inteligência de mercado da rede Campifarma, chegou-se a conclusão que este modelo estava apresentando resultados bastante coerentes. Assim, os analistas da rede Campifarma contrataram a elaboração de uma nova funcionalidade no nosso produto, o OnMaps, para a exibição dos resultados do modelo validado na aplicação.

Após a criação desta nova *feature*, os analistas da rede Campifarma poderão clicar em qualquer ponto do nosso mapa e a aplicação abrirá um pop-up no qual será exibida a predição de faturamento de uma eventual loja que será aberta naquele ponto. Além do valor em si, o pop-up deverá exibir, a quantidade de farmácias que estão bem próximas ao ponto clicado e, assim, se existe possibilidade de conversão de bandeira naquele ponto. Em caso positivo, estes pontos devem ser qualificados como uma unidade de uma grande rede de farmácias ou uma unidade de um pequeno varejista farmacêutico.

Arquivos

Para a entrega deste desafio técnico você precisará de 4 arquivos, que serão descritos a seguir:

- **pois.csv**: Arquivo que contém os pontos de interesse que impactam no faturamento das unidades da rede Campifarma. Este arquivo possui as seguintes colunas:
    - id: Identificador do ponto (int),
    - latitude: Latitude do ponto (float),

    - longitude: Longitude do ponto (float),
    - tipo_POI: O tipo do ponto de interesse, ou seja, se ele é uma lotérica, uma agência bancária, uma escola, uma própria unidade da rede Campifarma, ou um dos dois tipos de concorrentes da mesma.
- **faturamento.csv**: Arquivo que contém os dados de faturamento médio mensal de cada unidade da rede Campifarma. Suas colunas são:
    - id: Identificador do ponto. Este id é o mesmo do arquivo pois.csv.
    - faturamento_medio_mensal: O valor, em reais do faturamento médio mensal de cada unidade da rede de farmácias Campifarma.
- **campinas.wkt:** Arquivo contendo a geometria do município de Campinas. Detalhes sobre este formato de arquivo e a sua manipulação podem ser obtidos em: https://medium.com/geo-tech/manipulando-dados-geoespaciais-em-python-9fe21dda5894.
- **modelo_campifarma.ipynb**: Notebook no qual o Cientista de Dados fez o processo de criação do modelo a ser colocado em produção. **Atenção**: você NÃO deve treinar um novo modelo ou testar novas *features* para melhorar as métricas alcançadas com o trabalho do Cientista de Dados. Esperamos que o modelo obtido ao final no notebook seja o mesmo que será colocado em produção por você.

Os arquivos se encontram no seguinte *bucket* público da AWS: “s3://geofusion-insights-public/”.

Para baixá-los você precisará de ferramentas como a awscli (https://aws.amazon.com/pt/cli/).

Considerações técnicas

Esta nova feature deverá ser implementada na forma de um serviço. Para a criação dele, nossa equipe de Insights deverá trabalhar em conjunto com

a equipe de Engenharia, responsável pelo produto OnMaps. Como aqui na Geofusion adotamos a arquitetura de *microservices*, este serviço de predição de faturamento da rede Campifarma deverá ser um serviço REST, que será adicionado à nossa plataforma. A linguagem a ser utilizada deverá ser, necessariamente, a linguagem python (versão 3), pois ela é a linguagem padrão das aplicações do time de Insights.

Após conversas com a equipe de front-end da Geofusion, ficou acordado que o endpoint da sua aplicação será no seguinte padrão:

http://{*nome_do_servico*}:{*porta_do_servico*}/predict/lat={*lat*}&lng={*lng*}

E o retorno seria em formato json, também no seguinte padrão:
{
“latitude”: valor da latitude passada como parâmetro (float),
“longitude”: valor da longitude passada como parâmetro (float),
“predicao”: valor predito pelo modelo construído pelo cientista de dados para o ponto passado como parâmetro (float),
“n_grandes_concorrentes”: número de grandes concorrentes à menos de 50 metros do ponto passado como parâmetro (int),
“n_pequeno_varejista”: número de pequenos concorrentes à menos de 50 metros do ponto passado como parâmetro
}

Caso a latitude e longitude fornecida não esteja contida no município de Campinas sua aplicação deve retornar qualquer valor negativo para os campos “predicao”, “n_grandes_concorrentes” e “n_pequenos_concorrentes”.

Note que sua aplicação deverá responder à qualquer ponto que esteja contido no município de Campinas, e não somente aos pontos definidos no arquivo **pois.csv**. Seguem exemplos de alguns pontos e dos valores esperados. Observe que eles podem ser utilizados para validar o retorno da sua aplicação.

Exemplo 1:

Chamada:
http://localhost:5000/predict/lat=-22.8232257917&lng=-47.0758807513
Resposta esperada:


```
{
    "latitude":-22.8232257917,
    "longitude":-47.0758807513,
    "predicao":96040.6114295958,
    "n_grandes_redes":0.0,
    "n_pequeno_varejista":1.0
}
```


Exemplo 2:
Chamada:
http://localhost:5000/predict/lat=-22.8053524424&lng=-47.0064121403

Resposta esperada:


```
{
    "latitude":-22.8053524424,
    "longitude":-47.0064121403,
    "predicao":78410.6259787695,
    "n_grandes_redes":0.0,
    "n_pequeno_varejista":0.0
}
```


Exemplo 3:
Chamada:
http://localhost:5000/predict/lat=-22.982356215&lng=-46.9112167395

Resposta esperada:


```
{
    "latitude":-22.982356215,
```

```
        "longitude":-46.9112167395,
        "predicao":-1,
        "n_grandes_redes":-1,
        "n_pequeno_varejista":-1
}
```


Entregáveis

Você deverá publicar seu projeto no GitHub de forma anônima, ou seja, o projeto deverá ser **PRIVADO**. Após a sua criação, você deverá compartilhá-lo com o usuário “geofusion-insights” do GitHub. Neste projeto deverão estar contidos, além dos arquivos necessários para rodar a sua aplicação, dois arquivos extras:

1. **README.md**: um arquivo contendo as principais decisões técnicas tomadas por você. Note que este arquivo deve conter, também, as instruções necessárias para rodar a sua aplicação.
2. **respostas.txt**: arquivo contendo as respostas sucintas às seguintes perguntas:
    a. Parabéns! Sua aplicação está agora em produção, e, assim, você passa a ser responsável por ela! Como você garantirá que ela está respondendo conforme o esperado?
    b. É possível que, com o tempo, este modelo da Campifarma diminua a sua performance de predição? Em caso positivo, porque isso ocorre e como você solucionaria esse problema?
    c. Na sua visão, qual é a diferença de responsabilidades e de entregas de um Engenheiro de Machine Learning e de um Cientista de Dados?
    d. A Campifarma cresceu muito após a utilização do modelo que foi colocado em produção por você. Assim eles desejam expandir para todo o Brasil, e para o seu estudo de expansão desejam prever o faturamento em cada esquina o país, o que compreende por cerca de 10 milhões de pontos. Como você escalaria o seu serviço para responder à estas 10 milhões de requisições?

Sugestões

No nosso time fazemos verificações do código para garantir a qualidade do mesmo, quando colocado em produção. Especificamente, utilizamos o pylint (https://www.pylint.org/) e, em suas configurações default, não aprovamos código para produção que não alcance a nota 9.0 no mesmo. Ademais, procuramos utilizar as boas práticas de programação em python descritas neste artigo da ThoughtWorks (https://www.thoughtworks.com/insights/blog/coding-habits-data-scientists). Finalmente, adotamos o *Cookiecutter Datascience* (https://drivendata.github.io/cookiecutter-data-science/) para a organização dos nossos projetos. Maiores detalhes sobre o *Cookiecutter Datascience* e sua utilização podem ser vistos aqui: https://medium.com/@rrfd/cookiecutter-data-science-organize-your-projects-atom-and-jupyter-2be7862f487e.

Além disso, todos os nossos serviços estão implantados no Kubernetes, e para a conteinerização das nossas aplicações utilizamos o docker.

Entendemos que a manipulação de dados geoespaciais é algo muito específico do nosso negócio e que, por isso, você pode encontrar dificuldades nas manipulações de dados de geometria. Entretanto, entendemos que você pode se basear no próprio código contido no notebook enviado, na documentação do shapely (https://shapely.readthedocs.io/en/latest/manual.html) ou no nosso artigo do Medium (https://medium.com/geo-tech/manipulando-dados-geoespaciais-em-python-9fe21dda5894).

Considerações finais

A sua capacidade de compreensão dos comandos deste desafio técnico também está sendo avaliada. Entretanto, entendemos que alguma parte pode não ter ficado clara, e, assim, sinta-se à vontade para entrar em contato conosco, caso considere que isso seja realmente necessário para a conclusão do desafio.

Pedimos que você não compartilhe este texto e os seus resultados com ninguém ou mesmo em projetos PÚBLICOS na Internet, como, por exemplo, o GitHub. Após a entrega pedimos que você comunique nossa equipe de recrutamento, para que o nosso time técnico inicie o processo de avaliação de sua entrega. O seu projeto será analisado e você receberá um retorno a respeito do mesmo assim que a análise for concluída.

A partir do envio deste arquivo você terá 7 dias para terminá-lo. Caso precise de mais tempo entre em contato conosco. Agradecemos o seu interesse na Geofusion e esperamos que você se saia muito bem neste desafio técnico!