

Organização do Desafio Técnico do PD

Preparando a base BI:

- Na parte de montar BD's, posso preparar um conjunto de dados, uma versão do BD para alimentar uma rede neural, que ele esteja com boa formatação para uma rede neural (Introduction to DL for Tabular Data, fast.ai);
- Fazer uma cópia do BD original, fazendo a imputação de dados faltantes e validar a qualidade desse novo BD;
- Pegar dados de quantos usuários no Brasil de celular usam iOS ou Android, e deixar proporcional para avaliar a performance em termos de eficiência ou interface do aplicativo, em relação a ambas as plataformas, por que pode haver uma diferença de gosto estético e dinâmica do aplicativo de S.O. pra S.O., podem ter públicos diferentes, e se normalizar ambas e o tráfego em uma for muito diferente do outro, é possível que haja alguma coisa para mudar aí, que valha a pena investigar mais a fundo em relação aos aplicativos.

Sobre o pipeline:

- Por ser um Data Pipeline, o processo deveria ser eficiente, logo pode ser conveniente no final botar um wrapper sobre as funções mais custosas para otimizar elas.

Predições:

- Qual faculdade nos próximos meses vai ter mais assinantes?
- Podia pensar qual a tendência futura de uso de aparelhos na plataforma, para fazer análises de projeto e alocação de equipe.
- Qual faculdade nos próximos meses vai ter mais visualizações/acessos por mês?
- Qual a tendência de assunto a aumentar nos próximos meses?
- Seria possível deduzir o estado e cidade dos usuários para imputar os dados faltantes pela universidade, data de registro ou tipo de assinatura? (usando métodos de clusterização, NN etc)

Outras análises possíveis a partir da nova base (precisaria validar):

- Os alunos de quais universidades tendem a assinar o pacote premium, e outra, quais assuntos os alunos que assinam o pacote tendem a consumir?
- Das 10 principais faculdades/universidades, quais os assuntos mais acessados, seguidos?
- Quais os assuntos mais consumidos?
- Ver a correlação entre faculdade x acessos semanais, faculdade x registro de usuário, faculdade x assinatura, estado x acessos mensais, estado x assinaturas; fazer uma visualização no Bokeh com o mapa do Brasil talvez, verde claro pro

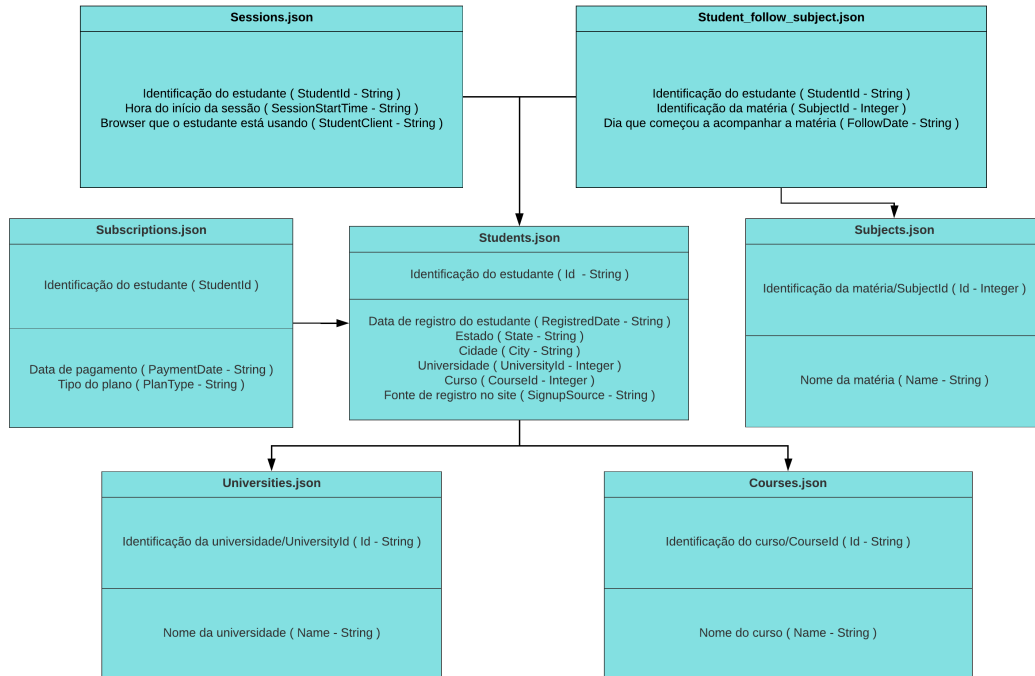
escuro ou azul claro pro escuro.

- Tenho que ver as características dos usuários que não botaram esses dados, e pensar como poderia fazer pra botarem.
- Estudantes do tipo X seguem matérias Y, qual a correlação mais forte nessas categorias?
- W pessoas da universidade R seguem matérias Y, qual a correlação mais forte nessa associação?
- O banco de dados vai de 2012 até 2017, pois o último iOS e Android são, respectivamente, o Android Oreo e o iOS 11. Seria possível tirar alguma relação dos aparelhos mais frequentes e o tipo de registro no site (Facebook/Google/E-mail)?
- Dos usuários cadastrados num intervalo de tempo X, quais as principais características Y dos que mais entraram no site? (tenho 60000 cadastros e 283941 entradas)
- Os dados de State e City têm muitos NaN, por que?

Sugestões:

- Dar sugestões para análises futuras, como usar Markov Logic Networks para extração de conhecimento e padrões.

Dados relacionais - Como estão estruturados



Tecnologias e bibliotecas usadas

Bibliotecas:

Visualização de dados:

- seaborn
- bokeh
- plotly
- matplotlib

Análise de dados e aprendizagem de máquina:

- fastai
- tensorflow
- sklearn

Otimização do código em geral:

- numba
- benchmarkit

Manipulação dos dados:

- pandas
- numpy
- scipy

Data pipelining:

- *Luigi, ou*
- Apache Airflow
- Docker pra rodar um deles

Referências

Encoders diferentes:

<https://towardsdatascience.com/smarter-ways-to-encode-categorical-data-for-machine-learning-part-1-of-3-6dca2f71b159>

One-hot encoding:

<https://hackernoon.com/what-is-one-hot-encoding-why-and-when-do-you-have-to-use-it-e3c6186d008f>

<https://medium.com/@arthurlambletvaz/one-hot-encoding-o-que-é-cd2e8d302ae0>

Pré-processamento de dados categóricos para Redes Neurais:

<https://www.fast.ai/2018/04/29/categorical-embeddings/>

<https://arxiv.org/pdf/1604.06737v1.pdf>

Difference coding pra lidar com dados categóricos:

<https://stats.idre.ucla.edu/spss/faq/coding-systems-for-categorical-variables-in-regression-analysis-2/#DIFFERENCE%20CODING>

<https://towardsdatascience.com/an-overview-of-categorical-input-handling-for-neural-networks-c172ba552dee>

Visualização de dados:

<https://matplotlib.org/3.1.1/index.html>

<https://www.analyticsvidhya.com/blog/2019/09/comprehensive-data-visualization-guide-seaborn-python/>

<https://seaborn.pydata.org/tutorial/categorical.html>

<https://d3js.org> (busca o d3js pra python)

<https://stackabuse.com/python-bokeh-library-for-interactive-data-visualization/>

<https://towardsdatascience.com/a-complete-guide-to-an-interactive-geographical-map-using-python-f4c5197e23e0>

<https://datascienceplus.com/seaborn-categorical-plots-in-python/>

<https://seaborn.pydata.org/generated/seaborn.countplot.html>

<https://seaborn.pydata.org/generated/seaborn.lineplot.html>

<https://dev.to/nexttech/how-to-perform-exploratory-data-analysis-with-seaborn-29eo>

<https://plot.ly/python/>

Para análise de séries temporais:

<https://stackoverflow.com/questions/16286991/converting-yyyy-mm-dd-hhmmss-date-time>

Manipulação de dados:

<https://numpy.org/devdocs/>

<https://www.geeksforgeeks.org/best-python-libraries-for-machine-learning/>

<https://www.youtube.com/watch?v=XMjSGGej9y8>

<https://pandas.pydata.org/pandas-docs/stable/reference/api/pandas.DataFrame.join.html>

https://matplotlib.org/api/pyplot_api.html?highlight=plot_date#matplotlib.pyplot.plot_date

<https://pandas.pydata.org/pandas-docs/stable/reference/api/pandas.DataFrame.iloc.html>

<https://stackoverflow.com/questions/9758450/pandas-convert-dataframe-to-array-of-tuples>

Benchmark pra otimização:

https://nbviewer.jupyter.org/github/vgrabovets/benchmark/blob/master/notebooks/benchmark_examples.ipynb

AutoEncoders pra fazer análise de dados categóricos:

<https://nbviewer.jupyter.org/github/donnemartin/data-science-ipython-notebooks/blob/master/deep-learning/keras-tutorial/3.1%20Unsupervised%20Learning%20-%20AutoEncoders%20and%20Embeddings.ipynb>

NN em tensorflow para DC:

https://nbviewer.jupyter.org/github/donnemartin/data-science-ipython-notebooks/blob/master/deep-learning/tensor-flow-examples/notebooks/2_basic_classifiers/nearest_neighbor.ipynb

Validação cruzada para séries temporais e outros para generalizar o modelo:

https://scikit-learn.org/stable/modules/cross_validation.html#leave-one-out-loo

Imputação de dados faltantes:

<http://www.scielo.br/pdf/csp/v25n2/05.pdf>

http://bibliotecadigital.fgv.br/dspace/bitstream/handle/10438/24742/dissertacao_glauco_azevedo.pdf?sequence=1&isAllowed=y

<https://www.tandfonline.com/sci-hub.tw/doi/abs/10.1080/00223891.2016.1252382>

<https://link.springer.com/sci-hub.tw/article/10.1007/s11162-014-9344-9>

Pré-processamento:

<https://scikit-learn.org/stable/modules/preprocessing.html#preprocessing>

MLN's para tomada de decisão e extração de conhecimento:

<https://www.datasciencecentral.com/profiles/blogs/markov-logic-networks-for-better-decisions>

<https://www.datanami.com/2018/07/03/can-markov-logic-take-machine-learning-to-the-next-level/>

Feature selection e correlação:

<https://towardsdatascience.com/feature-selection-correlation-and-p-value-da8921bfb3cf>

Pré-modelagem e extração de features:

<https://www.youtube.com/watch?v=V0u6bxQOUJ8>

Como fazer uma análise de segmentação de clientes:

<https://www.youtube.com/watch?v=0srjdRDh99Y>

<https://www.youtube.com/watch?v=4NDORb4HBkw>

Modelagem de dados:

<http://agiledata.org/essays/umlDataModelingProfile.html>

Para montar o Data pipeline:

<https://towardsdatascience.com/data-pipelines-luigi-airflow-everything-you-need-to-know-18dc741449b7>

<https://labs.getninja.com.br/usando-luigi-para-lidar-com-pipelines-de-tarefas-em-lote-batch-jobs-34544ab6cf16>

https://www.youtube.com/watch?v=ymF2R_tY1f8
<https://examples.dask.org/applications/prefect-etl.html>
<https://github.com/dask/dask-tutorial>
<https://medium.com/data-hackers/primeiros-passos-com-o-apache-airflow-etl-fácil-robusto-e-de-baixo-custo-f80db989edae>

Para fazer uma análise de séries temporais:

<https://datascience.stackexchange.com/questions/54138/how-can-time-series-analysis-be-done-with-categorical-variables>
<https://discuss.analyticsvidhya.com/t/time-series-forecasting-with-categorical-variables/7489>

Self-organizing maps para aprendizagem não-supervisionada

<https://towardsdatascience.com/self-organizing-maps-ff5853a118d4>

Final - Otimizando pandas com Numba/Cython:

https://pandas.pydata.org/pandas-docs/stable/user_guide/enhancingperf.html

Perguntas para um DataScientist:

<https://spin.atomicobject.com/2015/02/12/central-limit-theorem-intro/>
https://en.wikipedia.org/wiki/Accuracy_paradox
https://en.wikipedia.org/wiki/Precision_and_recall
<https://searchbusinessanalytics.techtarget.com/definition/data-sampling>
<https://365datascience.com/explainer-videos/#sql>
<https://towardsdatascience.com/understanding-random-forest-58381e0602d2>
<https://www.springboard.com/blog/data-science-interview-questions/>