# Hackathon JPA Agro 2021

## Data Science Research Group - DSRG
## Universidade Federal de Lavras - UFLA

# Identificação da equipe

**Integrante 1:** Aline Rodrigues Guimarães de Oliveira.
**Integrante 2:** Adriano Domingos Goulart.
**Integrante 3:** Ítalo Della Garza Silva.

# Descrição da solução

## 1. Entendimento do negócio

Embora, inicialmente, o grupo tenha focado na etapa de entendimento dos dados em si - verificando correlações e estatísticas contidas - houve um breve momento em que foram pesquisados dados e contextos externos como série histórica do preço da laranja e valor do Dólar ante o Real no período analisado. Embora tenha sido encontrado alguns dados que pareciam ter potenciais, por tempo, relação de prioridade e dificuldade de relacionar os dados no modelo preditivo, foi optado por não integrar estes aos dados.

## 2. Pré-processamento dos dados

Os dados estavam dispostos em sequência de dias, contendo o valor e o referente tipo de produto. O tipo de produto foi apagado pois possuía valor único em toda a base de dados. Foram identificados alguns dias inexistentes na sequência de datas. Porém, ao completar com zeros os valores faltantes verificou-se grande piora nos resultados. Logo, para esse aspecto, os dados nulos foram deletados da tabela. Foi efetuada a normalização dos dados para otimizar o treinamento do modelo LSTM, após a predição os dados sofreram a transformação inversa para retornar ao valor original.

## 3. Enriquecimento dos dados

Sobre o banco de dados disponibilizado, foram utilizados os dados de *negotiation_date* e *sold_price* - isso porque os dados se relacionam como uma uma série temporal - no caso relacionando as vendas e a data de sua negociação. O primeiro foi utilizado de forma indireta, já que pela natureza dos dados, sua maior vantagem era gerar uma ordem para os dados. Já o valor *product* foi desconsiderado porque continha um valor constante e por isso não agregava nenhum valor diferencial a cada instância.

## 4. Modelos

Vário modelos foram testados com diversas taxas de aprendizagem e número de repetições, o que aprensentou melhor RMSE foi mantido. O melhor modelo encontrado para

esta solução, foi uma LSTM com apenas uma camada oculta de 350 neurônios. Para o treinamento, foi utilizado o otimizador *Adam* com uma taxa de aprendizado fixada em 0,001 e 1.000 repetições. Para a melhoria na inicialização dos pesos, foi utilizada a técnica de (Glorot, Bengio). Os *batches* de treinamento foram sendo obtidos aleatoriamente a cada passo do treinamento.

## 5. Avaliação da solução

Sobre a divisão dos conjuntos de dados de treinamento e de teste, primeiramente é importante ressaltar que os membros da equipe tiveram a percepção que seria interessante utilizar de uma rede recorrente por se tratar de uma série temporal - supomos isso pela natureza do desafio e dos dados. Por isso, na divisão de dados, foi utilizada uma técnica para dividir os dados em períodos de 30 (pensando em um mês), e, além disso, utilizando um dado para prever o próximo. Isto é, considerando que $X$ é o conjunto de treinamento e $y$ o conjunto de respostas do treinamento, e, considerando que estes estão ordenados de forma temporal - $y_0$, por exemplo, deve ter o valor de $X_1$.

Além disso, foram utilizados os últimos 31 dados - 1 período - para fazer o teste da base de dados. Nas variáveis de teste foram aplicadas as mesmas divisões de dados das bases de treinamento. Foram feitas 10 execuções de todo o treinamento, sendo que os *batches* foram obtidos aleatoriamente a cada passo de cada execução do treinamento. A geração do arquivo com as predições para 30 dias futuros foi feita fazendo o uso dos últimos 30 dias do conjunto de treinamento. A cada iteração, o resultado era inserido no fim do conjunto de treinamento para a geração do novo resultado, e assim sucessivamente até que fosse gerada toda a predição final.

Além do RMSE (*Root Mean Square Deviation*), que será utilizado na avaliação, foi utilizada a métrica MAE (*Mean Absolute Error*) ao longo de execuções com o modelo LSTM. A equipe testou outros métodos de Machine Learning, como uma RNN Simples (*Recurrent Neural Network*) e também um modelo SARIMA (*Seasonal Autoregressive Integrated Moving Average*) pelo fato da série temporal apresentar comportamento sazonal, porém seus resultados foram piores que o LSTM. A implementação do RMSD utilizada foi a disponibilizada, já o MAE foi importado da biblioteca *sklearn* - função *mean_square_error*.

Para o conjunto de treinamento, o RMSE médio ficou em 54.89 e o MAE médio ficou em 34.67.

# Referências

Glorot, Xavier e Bengio, Yoshua (2010). Understanding the difficulty of training deep feedforward neural networks. Journal of Machine Learning Research - Proceedings Track. 9. 249 – 256.