**PREDICT THE PROBABILITY OF OCCURENCE OF AN EVENT**

PREPARATION OF THE DATA

* Libraries: autoviz, dataprep,
* Skewed(log ,1+log)
* Outliers: everything outside 3 std from the mean , instead of removing them you can replace with the median or the mean
* Imbalance ? How to fix it (smothee? Do some research)
* Time? How to account for it? Shall we build new features? YES

MODELS TO USE

* XGBOOST WITH HYPEROPT -> our event is 0s or 1s
* model.predict\_probab
* XGB in h20 and dallax
* LIME for interpretability of the models
* Out of fold cross validation

ALTERNATIVE APPROACHES FOUND ONLINE TO PREDICT THE PROPENSITY TO BUY OF A CUSTOMER

* Based on the results of your EDA, you can choose a model to predict the likelihood of a customer buying a product. A popular choice is the Beta-Geometric Negative Binomial Distribution (BG-NBD) model, which models customers' repeat purchasing behavior using the Gamma and Beta distributions [towardsdatascience.com](https://towardsdatascience.com/bayesian-customer-lifetime-values-modeling-using-pymc3-d770676f5c06).
* Here is how you can build your approach with a frequentist approach <https://www.phind.com/search?cache=c89b6wi2rpdoynlhuixdr7qw>
* Here's a simple example of how you might implement a BG-NBD model using PyMC3: (if you know some assumption about your data then a good one would be to use bayesian)
* **import pymc3 as pm**
* **with pm.Model() as model:**
* **r = pm.Gamma('r', alpha=0.5, beta=1)**
* **p = pm.Beta('p', alpha=1, beta=1)**
* **y = pm.Binomial('y', n=n, p=p, observed=data)**
* **step = pm.Metropolis()**
* **trace = pm.sample(10000, step=step)**
* Remember to make the series stationary before applying the model <https://www.phind.com/search?cache=xvqsfjk1q31g8gzv78hkoark>
* So we could have this situation: 
* Or we could have a situation where I want to predict whether a consumer prefers one product A over B/C/D (could it be A/B testing?) Do some research.