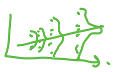
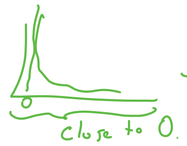
BA & ML – Analytics Cup Plan

1. Check R^2, it should be high. Particularly Multiple R-Squared.
2. Check VIF (Variance Inflation Factor)
   - → If VIFs are ↑↑, check correlation coefficients
   - → If some columns are correlated drop one by one, check VIF
   - → If 2 variables are correlated at a rate greater than .6, .7, .8 try dropping the least theoretically important of the two.
3. Check p-values [Pr(>|t|)] of columns, they should be significant (***)
4. Apply Glescher Test to see if there is heteroscedasticity in a particular column .
   - → Needs normally distributed residuals.
5. Apply White Test to see if there is heteroscedasticity.
   - → Not needs normally distributed residuals.
6. Use Durbin-Watson (DW) Method to see if there is autocorrelation in the column.

7. If you see  => You can apply log transformation.

→ If some X has quadratic relationship with Y, you can include I(y^2) in your model.

8. If data is "panel data", consider testing endogeneity and use either Random Effects Model or Fixed Effects Model.

9. You can use Wald Statistics and remove irrelevant variables.

10. If you use Logistic Regression, use Wald-test for significance (similar to t-test for Linear Regression)

11. If there is a "reverse" ordinal column, like "rank" you can use as.factor(rank) in your model.

12. If you use Logistic Regression, use McFad for understanding explanatory power of covariates (similar to R^2)

13. Pay attention to data types! (Nominal, Ordinal, Interval, Ratio). You can sometimes use ordinal as numeric (similar to ratio).

14. If there are many attributes and attributes seem to be independent and equally important, you can use Naïve Bayes Classifier.

15 If you use Decision Trees, first only consider attributes with greater than information gain, then compare them on gain ratio.

16 If you use a tree-based method, consider binning numeric attributes so that they can be treated as nominal attributes.

17 Especially in tree-based methods, you can add an additional attribute as "attribute_a > attribute_b".

Data Preprocessing

COLUMNS REQUIRING "ONE HOT ENCODING" ONLY:

- **PRICE_LIST** -> ONE HOT ENCODING #DONE
- **TECH** -> ONE HOT ENCODING #DONE
- **OFFER_TYPE** -> ONE HOT ENCODING #DONE
- **BUSINESS_TYPE** -> ONE HOT ENCODING #DONE
- **SALES_OFFICE**: ONE HOT ENCODING #DONE
- **SALES_BRANCH**: ONE HOT ENCODING #DONE
- **OWNERSHIP:** ONE HOT ENCODING
- **SALES_LOCATION**

OTHER COLUMNS:

- ~~**END_CUSTOMER** -> HAS_END_CUSTOMER~~
- ~~**ISIC** -> HAS_ISIC~~
- ~~**COSTS_PRODUCT_A** to **COSTS_PRODUCT_E**:~~
  - ~~COSTS_PRODUCT_* -> INCLUDED_COSTS_PRODUCT_*~~
  - ~~TOTAL_COSTS_PRODUCT: SUM OF ALL COSTS_PRODUCT_*~~
- ~~**COUNTRY_CODE**: Convert to BINARY~~
- **REV_CURRENT_YEAR**.1 and **REV_CURRENT_YEAR**.2:
  - ~~**REV_CURRENT_YEAR.1** -> EURO (with a fixed exchange rate)~~
  - ~~**REV_CURRENT_YEAR.2** -> EURO (with a fixed exchange rate)~~
  - ~~DROP ONE OF THEM: REV_CURRENT_YEAR.1 or REV_CURRENT_YEAR~~
  - ~~0 <- not 0~~
  - ~~New column PREV_YEAR_PERCENTAGE_INCREASE: ((REV_CURRENT_YEAR - * REV_CURRENT_YEAR.2)/REV_CURRENT_YEAR)*100)~~
- ~~**CREATION_YEAR**: Without null values, look at correlation with the target. If important, then think about filling missing values.~~
  - ~~Extract only the year.~~
  - ~~Calculate how long since CREATION_YEAR~~
- ~~**OWNERSHIP:**~~
  - ~~NA <- No information:~~
  - ~~Without null values, look at correlation with the target. If important, then think about filling missing values or no information.~~
  - ~~Without "no information" values, look at correlation with the target. If important, then think about filling missing values or no information.~~
  - ~~Change it to one-hot encoding.~~

Additional Advice from Stefan Heidekrüger

-> Log transformations: log+1

-> Create a BAC pipeline

-> do not make data leakage
-> different customer ids in train and validation set
-> If Nominal columns are majority: tree-based solutions are better
-> If numerical columns are majority: logistic solutions are better
~~-> In date columns: you can extract Similar to unix timestamp, you can convert dates to~~
-> LogReg,Random Forest: For most cases one-hot encoding is dealt implicitly
-> If necessary: After one-hot encoding. Last 5% or last 10% to some "others"
-> not more than 150 columns
-> log transformations are kinda outlier prevention

Filling Missing Values

- **CREATION_YEAR**
- **REV_CURRENT_YEAR.1**
- **REV_CURRENT_YEAR.2**
- **REV_PERCENTAGE_INCREASE**
- **REV_CURRENT_YEAR**.1 and **REV_CURRENT_YEAR**.2: If both zero or NA, before replacing, be sure that it is converted to euro, then replace with mean
- **OWNERSHIP**
- **OWNERSHIP_NO_INFO_AS_NA**
- **SALES_LOCATION** -> ONE HOT ENCODING
    - In Test set and has no SALES_LOCATION
        - SALES_LOCATION: NA TEST_SET_ID: 12359, Random Sales location in CH (?)
        - SALES_LOCATION: NA TEST_SET_ID: 16396, Random Sales location in CH (?)

Feature Importance/Check

- Correlation matrix between all columns
- R library for ranking which features are statistically significant
    - `library(Boruta) -` https://www.machinelearningplus.com/machine-learning/feature-selection/
    - `VIP model about feature importance`
    - `Shap values - expensive to compute`

- For numerical values, plot independent values with respect to the target, then consider applying log transformation to the column with many 0s.
- Isolate numerical values and try PCA
- Apply Logistic Regression
    - Check Variance Inflation Score
    - Apply McFad to mimic R^2.

    - For significance, use wald test
- You can try Random Forest, Gradient boosted trees etc.

! Remember to set the seed before doing anything ! :)

+ Joining, pre-processing
+ Extreme Values, Missing values
+ Data Augmentation: Feature extraction, combine features, separate features
+ Feature Importance, Explanatory

+ **Models:**
  - Ensemble: Majority Voting, bagging, boosting
  - Start with: simple logistic regression
  - Decision Trees
  - Naive Bayes
  - Random Forest
  - Neural Networks
  - AutoML: to narrow down options

Split validation: split by customer

Task distribution:

Weekly Plan

Week1:
Data Preprocessing

Week2:

Week3: