# 1 Data selection and binarization

As my big home work assignment, I choose Neural FCA. I have found 3 datasets with mostly categorical data, appropriate for FCA methods.

1. The Banking Dataset Classification of 33000 rows.
2. A dataset of Bike Buyers of 1000 rows.
3. A dataset of the car acceptibility (1700 rows).

### 1.0.1 Banking Dataset Classification

The first dataset contains the economical and commercial information about the bank clients and their acceptability during current marketing campaign:

1. **age** - age of a person
2. **job** - type of job
3. **marital** - marital status
4. **education** - type of graduation
5. **default** - has credit in default?
6. **housing** - has housing loan?
7. **loan** - has personal loan?
8. **contact** - contact communication type ('cellular','telephone')
9. **month** - last contact month of year
10. **day_of_week** - last contact day of the week
11. **duration** - last contact duration, in seconds
12. **campaign** - number of contacts performed during this campaign and for this client (includes last contact)
13. **pdays** - number of days that passed by after last contact from a previous campaign (999 if was not previously contacted)
14. **previous** - number of contacts performed before this campaign and for this client
15. **poutcome** - outcome of the previous marketing campaign ('failure','nonexistent','success')
16. **y** - TARGET outcome of the current marketing campaign

Features 'job', 'marital' ,'education' ,'default', 'housing' and 'loan' can have values of uncertainity. I decided to clear the dataset from any uncertain values and got 24305 rows of 33000 The chosen binarized features are the following:

1. **high_educ** - professional skills (job = university/professional_course)
2. **has_job** - has a job (not a student/unemployed/retired
3. **married** - not single of divorced/widowed
4. **default**
5. **housing**
6. **loan**
7. **cell_cont** - contacts by cellular phone

8. **before_con** - there were contacts performed before this campaign
9. **current_con** - there were contacts performed during this campaign
10. **long_dur** - the last contact was longer than the mean contact with negative result
11. **poutcome** - success of the previous campaign
12. **y** - TARGET

Features 'default', 'housing', 'loan', and 'y' remained the same as in the dataset.

#### 1.0.2 Bike Buyers

The dataset describes potential buyers of a bike with their economic status and commuting information.

1. **ID**
2. **Marital Status**
3. **Gender**
4. **Income**
5. **Children** - number of children the customer has
6. **Education** - type of the customer graduation
7. **Occupation** - professional level of the customer's job
8. **Home Owner** does the customer have a real estate?
9. **Cars** - number of cars the customer has
10. **Commute Distance** - distance to work (several interval of miles)
11. **Region** - Europe/North America/ etc.
12. **Age**
13. **Purchased Bike** - target value of buying a bike

According to the task priorites, the chosen features for binarization and the further FCA are: 'Income', 'Cars', 'Commute Distance', 'Gender'. The binarized features are:

1. Income 10000-60000, Income 60000-170000 - binary distribution of the income
2. Cars 0, Cars 1-4 - binary distribution of the number of cars
3. Commute Distance 0-1/1-2/2-5/5-10/10+ Miles - bined distribution of the commute distance
4. Gender Male
5. Purchased = **Purchased Bike**

### 1.0.3 Car Acceptability

The last dataset contains information of car parameters and its overall market acceptability. Most of features are measures of car main qualities, but designed as categorical and has values 'low', 'medium', 'high and 'very high' (called 'vhigh').

1. **Buying price** - from low to very high
2. **Maintenance price** - from low to very high
3. **No_of_Doors** - 2,3, 4 OR 5+
4. **Person_Capacity** - 2, 4 or more
5. **Size_of_Luggage** - small, medium or big
6. **Safety** - low, mdium or high
7. **Car_Acceptability** - target value - 'unacc', 'acc', 'good', 'very good'.

To minimize amount of unique values in columns the dataset was reduced to rows without 'vhigh' values, while acceptablity was divided into two values: $0 = unacc$ and $1 = acc - vgood$. The datset was also stripped from features 'No_of'_Doors', 'Person Capacity' and 'Size of Luggage'. The remained features were binarized for each unique value.

Final set of features is the following:

1. **Buying Price high/low/med**
2. **Maintenance Price high/low/med**
3. **Person Capacity 2/4/more**
4. **Safety high/low/med**
5. **Car Acceptability**

## 2 Baselines

The classification on the data selected in the previous homework was performed using the following 4 methods:

1. Decision tree
2. Random forest
3. xGboost
4. k-NN classifier

The results presented on the figure 1. It is seen that every model performs on nearly same perfomance on the same data. However, the performance is completely different on different datasets, which can show the quality on binarization. The notebooks of binarization and classification are presented here.

| | model | dataset | params | accuracy | f1_score |
|---|---|---|---|---|---|
| 0 | DecisionTreeClassifier | 1 | {'max_depth': 15, 'min_samples_split': 4} | 0.714600 | 0.730883 |
| 1 | RandomForestClassifier | 1 | {'max_depth': 2, 'min_samples_split': 2, 'n_es... | 0.730400 | 0.747931 |
| 2 | XGBClassifier | 1 | {'learning_rate': 1.1, 'max_depth': 3, 'n_esti... | 0.730400 | 0.750554 |
| 3 | KNeighborsClassifier | 1 | {'n_neighbors': 7} | 0.620800 | 0.434890 |
| 4 | DecisionTreeClassifier | 2 | {'max_depth': 10, 'min_samples_split': 2} | 0.574000 | 0.512467 |
| 5 | RandomForestClassifier | 2 | {'max_depth': 5, 'min_samples_split': 7, 'n_es... | 0.595000 | 0.552535 |
| 6 | XGBClassifier | 2 | {'learning_rate': 0.7, 'max_depth': 3, 'n_esti... | 0.597000 | 0.554189 |
| 7 | KNeighborsClassifier | 2 | {'n_neighbors': 3} | 0.497000 | 0.499329 |
| 8 | DecisionTreeClassifier | 3 | {'max_depth': 10, 'min_samples_split': 2} | 0.917706 | 0.895381 |
| 9 | RandomForestClassifier | 3 | {'max_depth': 2, 'min_samples_split': 2, 'n_es... | 0.940317 | 0.927784 |
| 10 | XGBClassifier | 3 | {'learning_rate': 0.9, 'max_depth': 3, 'n_esti... | 0.942363 | 0.928916 |
| 11 | KNeighborsClassifier | 3 | {'n_neighbors': 3} | 0.925927 | 0.900744 |

Рис. 1: Results of the classification research: number of datasets are the following: 1 - Bank Classification, 2 - Bikes, 3 - Cars.

## 3 Neural FCA

The Neural FCA was performed on each dataset. As each dataset have different number of rows and different number of features, we use different train ratio on each dataset. It does not reduce the train quality, if the train size is proportionate to amount of possible concepts. For example: the 'bike' binarized dataset has 10 columns, but it's not independent features, as only one value available per each feature. As a result, the number of concept is $2 \times 2 \times 5 \times 2 = 40$, which much less than 800 objects took for training. On other hand, for densely binarized datasets (where each columns connect to a unique feature from original datasets), such as 'bank', the number of concepts is $2^1 1$, which makes training more difficult.

The number of concepts enough to cover all features is also highly different in each dataset - for 'bikes' dataset it was only 6, but for 'bank' datset even 30 concepts couldn't cover all features.

It is also important to balance the target value ratio in the train and the test data. For example, 'cars' and 'bank' datasets were highly unbalanced, so it was required to take equal amount of positive and negative objects deliberately. But the same time, it was crucial to have a sufficient number of both types of objects in the test area, to make sure that estimation of the quality will be fair.

The results of FCA training are presented on the fig. 2. It is clearly seen that FCA method performed slightly better both in accuracy and in f1-score than the common ML baselines.

| | dataset | train_ratio | concepts | f1_scope | accuracy |
|---|---|---|---|---|---|
| **bank** | 1 | 0.160000 | 30 | 0.759014 | 0.718333 |
| **bikes** | 2 | 0.800000 | 8 | 0.600000 | 0.640000 |
| **car_accept** | 3 | 0.411523 | 15 | 0.935484 | 0.958042 |

Рис. 2: Results of the classification research: number of datasets are the following: 1 - Bank Classification, 2 - Bikes, 3 - Cars.