# Dataiku Data Scientist Technical Assessment

Jules Boistard

*18/01/2022*

# Problem statement

**Goal** : identify characteristics associated with a person making more or less than $50k a year

**Population** : individuals living in the US (from 1994/95 US Census Bureau surveys)

**Dataset** :
- ~300k observations (homogenous groups of individuals
- 40 features (age, sex, race, education, field of work, type of job, *etc.*)
- + 1 target variable : low income (<50k) / high income (>50k)

0 1

## Binary Classification Problem

Predict income class as accurately as possible based on the given set of features

## Inferential model

Maintain explainability throughout data modelling

# Data cleaning

**Duplicates & conflicting instances** :

- Duplicates : same exact observations (target variable included)
- Conflicts : same features values but different target class

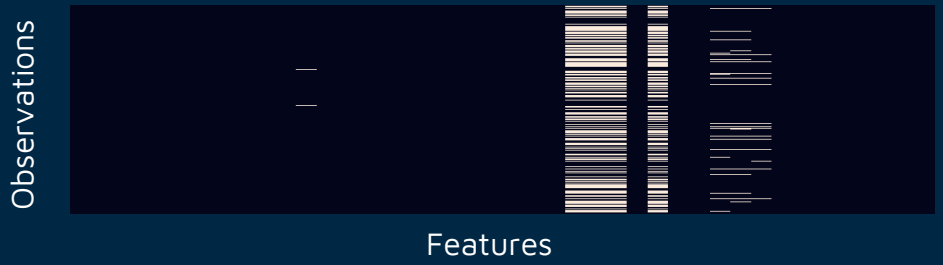
To pick between the two classes for conflicts, decision was made based on the highest total instance weight across all observations for each class

| | Train set | Test set | Total |
|---|---|---|---|
| Instances | 199 523 | 99 762 | 299 285 |
| Duplicates | 46 627 | 20 898 | 67 525 |
| Conflicts | 89 | 38 | 127 |
| Drop rate | 23.4% | 21.0% | 22.6% |

*Note that provided we had domain expertise and the ability to collect additional data, we could end up with extra features allowing us to tell these observations apart, and keep them for better predictive power during modelling*

**Missing values**

- 4 features had ~50% of missing values → dropped
- 5 features had smaller missing values rate → kept with imputation (most frequent class)




*(missing values are in white)*

# Distributions within target classes

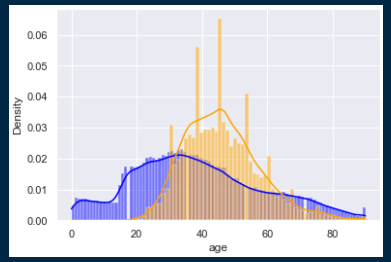
**Class imbalance issue** :

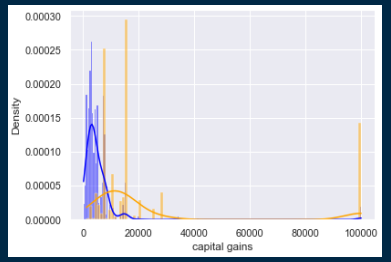
- « Low income (<50k) », the majority class, accounts for ~92% of all obs.
- Easier too look at variables distributions **withing each class**

## Notable distributions for numerical features

**Low income (blue)**
**High income (yellow)**

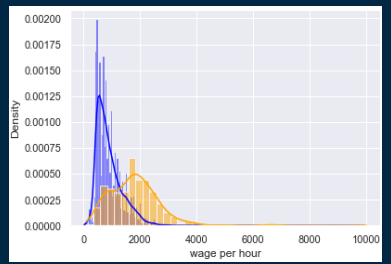
**Age** : children (under 18) belong exlusively to the "low income" class, as could be expected

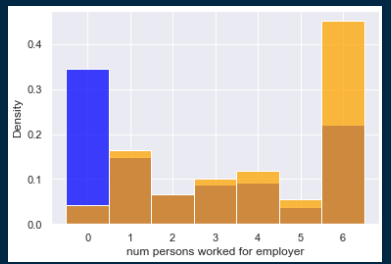



**Capital gains** : gains > 10k are almost exclusively belonging to the "high income" class

**Wage per hour**: wages > 2k are almost exclusively belonging to the "high income" class




**Nb of employees in company** : people without a job largely belong to the "low income" class (not very informative)

# Distributions within target classes

**Categorical features** : using chi2 tests, we can find the most relevant features for classification. Among the top 10, we can extract some useful insights

| | score |
|---|---|
| detailed industry recode | 78479.449 |
| detailed occupation recode | 8224.323 |
| hispanic origin | 7402.586 |
| education | 3969.421 |
| sex | 3534.538 |
| marital stat | 2495.073 |
| tax filer stat | 2054.010 |
| major industry code | 1458.384 |
| major occupation code | 1210.807 |
| detailed household and family stat | 1130.907 |

**Industry** : high incomes much more frequent in public administration and finance/insurance/real estate for instance

**Occupation** : high incomes much more frequent among professional specialties and executives/maangers

**Education** : high incomes much more frequent above Bachelor's Degree

**Sex** : female are underrepresented within the high income class

**Marital status**: > 75% of high income individuals are married (not divorced nor widowed)

**Householders** : much more frequent among the high income class than the low income class

# Data pre processing

1. **Missing values** :
   - Drop features with 50% missing values
   - Impute remaining missing values with most frequent category of feature
2. One hot **encoding** of categorical variables
3. Standard **scaling** for numerical features
4. **Resampling** with SMOTE to try and overcome class imbalance problem
5. **Additional features** based on available information :
   - **earnings estimate** based on wage, weeks worked, capital gains/losses and dividends from stocks
   - **company size** estimate based on segments in documentation
   - **tax amount estimate** based on earnings estimate and US federal tax rates and brackets for 1995

Two simple baseline models were used to evaluate pre-processing steps performance : logistic regression and decision tree

Based on the results, resampling was kept for linear model only, whereas extra features were kept for tree-based models only

# Model tuning and selection

**Scoring metric** : macro average f1 score accross both classes to account for class imbalance

**Competing models** :

- Random Forest (tree-based)
- Gradient Boosting (tree-based)
- Stochastig Gradient Descent (linear model)

**Note** : to maintain explainability throughout the modelling process, more complex models, such as neural networks, were set aside

**Selection** : grid search for hyper-parameter tuning combined with cross-validation

Best performing model :

**Gradient Boosting**

*(500 estimators, 0.1 learning rate and max depth of 5)*

| | Precision | Recall | F1-Score |
|---|---|---|---|
| **Low income (<50k)** | 95,62% | 98,21% | 96,90% |
| **High income (>50k)** | 68,93% | 45,95% | 55,86% |
| **Macro average** | 82,28% | 72,58% | **76,38%** |

# Key points

**Room for improvement** : the score is still rather low, espcially zooming in on the minority class (high incomes)

- Collect **additional observations** of the minority class to resolve balancing issue. We should not be that much limited by available data (as in a fraud detection problem), as >50k incomes should be common enough to gather a sufficient sample of observations.
- Perform more **feature engineering** with domain expertise and gathering of extra data if necessary
- Take a deeper dive into **feature importance**, for instance by computing Shapley values, after modelling
- Further investigate **dependencies in-between features** to avoid multicollinearity