Sberbank Data Science Contest

#	Участник	Общий балл	Задача A	Задача B	Задача C
77	mrk-andreev (DMIA)	358.8919	0.896297 (169.4049)	1.575759 (189.4871)	1.647985 (0.0000)
(11%)			14 (2%)	59 (8%)	194 (27%)

Задача A

- RobustScaler
- Много статистик для transaction

```
F2 = transactions.groupby('customer_id')
            .apply(lambda x : x.groupby('mcc_code')['amount'].sum())
            .unstack().fillna(0)
F2.rename(columns=lambda x: 'sum_'+str(x), inplace=True)
```

- Разделение transaction по признаку amount>0
- Слияние 3ех групп признаков
- XGBoost
- Слияение решений* с помощью rankdata

* мое + решение с форума

Статистики для transaction

1) Делим transations на: {∀ amount, amount>0, amount<0}
2) Для каждой группы вычисляем для групп mcc_code, tr_type, вычисляем sum, mean, std, max
3) Сколько дней между первой и последней транзакцией:
   days* = |max(days)-min(days)|

Итог: ~4000 признаков

Слияние решений

```
from scipy.stats import rankdata


r_ans = pd.read_csv('../data/raw/task1_solution_by_const.csv')

blended_submit = submit_data.copy()
blended_submit['gender'] = rankdata(r_ans['gender'].values) + rankdata(submit_data['gender'].values)
blended_submit.to_csv('../data/submits/sbm_' + ts + '_blended.csv', index=False)
```

Задача B

- XGBoost

```
k = 500
param = {
  'eta' : 0.2/float(k),
  'max_depth' : 5,
  'colsample_bytree' : 0.2,
  'min_child_weight' : 13,
  'gamma' : 14,
  'subsample' : 0.7,
  'objective' : 'reg:linear',
  'eval_metric' : "rmse"
}

clf = xgboost.train(param, dtrain, num_boost_round=100*k)
```