Report

Group A: Alexander Benz, Lukas Zaiser, Sven Weiß

Introduction and data

In the last 30 years, the dating approach has changed and has become increasingly difficult. The willingness to date has decreased, dating is too expensive and time consuming, we have too many (perceived) options to date someone and we struggle because of accepting too easily negative sex stereotypes. In the 19th century, a custom in the United States called New Year's Calling, was that on New Year's Day many young, single women would hold an Open House (a party or reception during which a person's home is open to visitors) on 1 January where they would invite eligible bachelors, both friends and strangers, to stop by for a brief (no more than 10–15-minute) visit. This custom was established with the term SpeedDating as a registered trademark by Aish HaTorah, who began hosting SpeedDating events in 1998.

10 years later, Fisman et al. conducted a survey regarding speed dating habits and collected 8,000 observations during his 2 – year observation in his paper Gender Differences in Mate Selection: Evidence from a Speed Dating Experiment. Because speed dating has become more and more interesting in the last few years and also through Corona a completely new dating approach has emerged, we wanted to analyse this dataset with the following questions in mind:

- **What are the most effective personal characteristics to achieve a match in opposite sex speed dating?**
  - A match may be a high like value (1 - 10, regression) or a positive match (1 or 0, classification)

The following hypotheses support our research question:

Null hypothesis:

- **There is no affection of having specific characteristics regarding match selection of the survey participants**
- **There is no correlation between shared interests, attributes and getting a match**

Hypotheses:

- **Survey participants who both have the specific characteristics same race and opposite gender tend to achieve more matches**
- **Survey participants with a higher income tend to achieve more matches than survey participants with a lower income**
- **Achieving matches because of having the same specific characteristics occur in both sexes**
- **Three weeks after the event, males called women more often**

Our dataset was pretty helpful in answering this and more questions, as there were a lot of helpful features:

We want to answer our research questions in 4 steps:

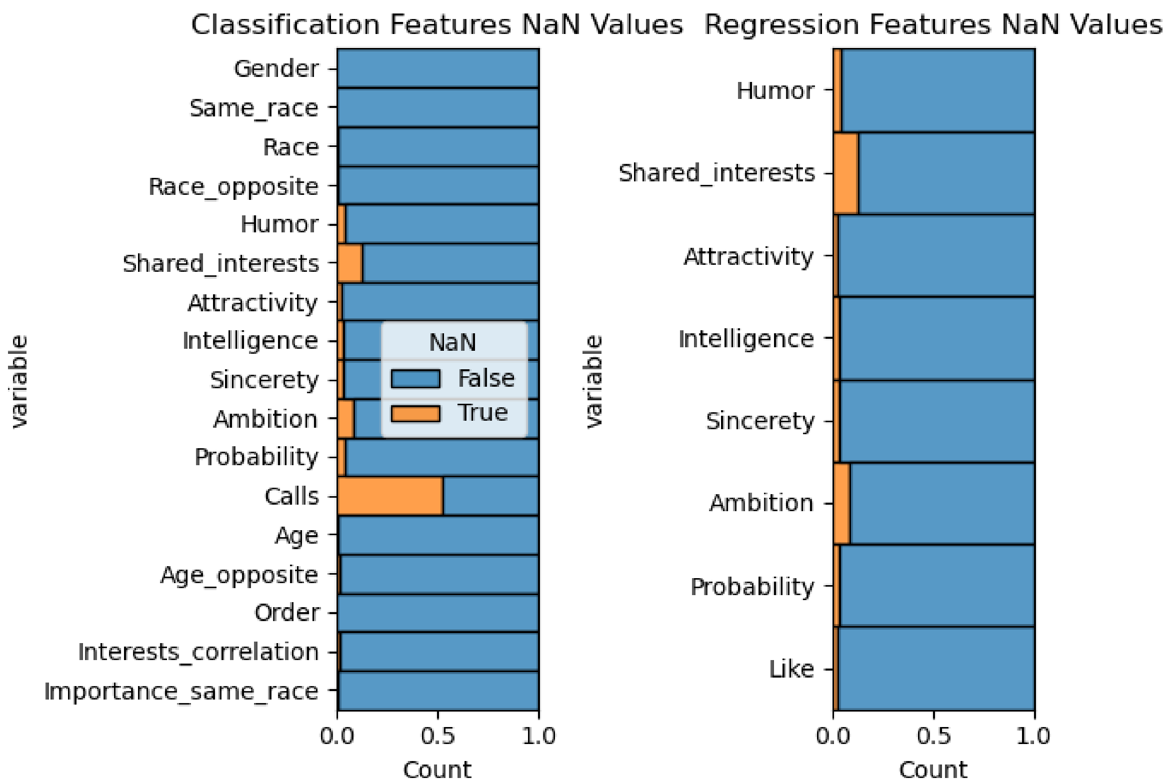
- Step 1 Importing the required libraries
- Step 2 Cleaning the dataset
- Step 3 Analyzing the dataset
- Step 4 Preparing the model
- Step 5 Analyzing the model

The main, effective variables we want to look at to answer our research questions are 'Match' (as our predictor variable for the classification) including the personal attributes/features and 'Like' for the regression. For all variables, we use descriptive terms in order to recognize them better. First, we want to analyze the importance of each personal attributes for achieving a match (classification) on the one hand and for the strength of a like (regression) on the other hand.

Data cleaning

For the calls variable, we assume that NaN had zero calls. This is of cause only an estimation. This value was collected after the events so not many answered this question. For all the other attributes we drop the NaN values because the ratio is rather small.

Overall, there are a lot of missing values for questions that were asked after the events like the *_2 and *_3 attributes, so we concentrated on the answers given at the events.




```
--- Regression ---
<class 'pandas.core.frame.DataFrame'>
Int64Index: 8378 entries, 0 to 8377
Data columns (total 8 columns):
```

```
 #   Column            Non-Null Count  Dtype
---  ------            --------------  -----
 0   Humor             8028 non-null   float64
 1   Shared_interests  7311 non-null   float64
 2   Attractivity      8176 non-null   float64
 3   Intelligence      8082 non-null   float64
 4   Sincerety         8101 non-null   float64
 5   Ambition          7666 non-null   float64
 6   Probability       8069 non-null   float64
 7   Like              8138 non-null   float64
dtypes: float64(8)
memory usage: 847.1 KB
None

--- Classification ---
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 6784 entries, 0 to 6783
Data columns (total 17 columns):
 #   Column                 Non-Null Count  Dtype
---  ------                 --------------  -----
 0   Gender                 6784 non-null   category
 1   Same_race              6784 non-null   category
 2   Race                   6784 non-null   category
 3   Race_opposite          6784 non-null   category
 4   Humor                  6784 non-null   float64
 5   Shared_interests       6784 non-null   float64
 6   Attractivity           6784 non-null   float64
 7   Intelligence           6784 non-null   float64
 8   Sincerety              6784 non-null   float64
 9   Ambition               6784 non-null   float64
 10  Probability            6784 non-null   float64
 11  Calls                  6784 non-null   float64
 12  Age                    6784 non-null   float64
 13  Age_opposite           6784 non-null   float64
 14  Order                  6784 non-null   int64
 15  Interests_correlation  6784 non-null   float64
 16  Importance_same_race   6784 non-null   float64
dtypes: category(4), float64(12), int64(1)
memory usage: 716.3 KB
None
```

Methodology

After cleaning our dataset and our initial exploratory data analysis, we can see the relationships between the respective outcome and possible predictors for each of the classification and regression.

For **regression** we use the following models:

- Linear Regression,
- Multiple Regression
- Lasso Regression
- XGBOOST Regression Models

The considered metrics for regression are:

- R2-Score
- Mean squared error
- Mean Absolute Error
- Root Mean Squared Error

For **classification** we use the following models:

- Logistic Regression

The considered metrics for logicstic regression are:

- Confusion Matrix
- Precision, Recall, Accuracy and F1 scores
- Precision-Recall curve
- ROC curve and AUC value

Model selection process for Classification: Besides logistic regression there are other types of classification algorithms like Naïve Bayes, Stochastic Gradient Descent, K-Nearest Neighbours, Decision Tree, Random Forest and Support Vector Machine. Since we need a machine learning algorithm which is most useful for understanding the influence of several independent variables on our single outcome variable, we use the Logistic Regression, which is modelling the probabilities describing the possible outcomes of a single trial.

For the Logistic Regression, we use the LogisticRegressionCV model. On default, this model includes a 5 cross fold validation with Stratified K-Folds so there is no need to do further training and validation. See the Scikit-Learn documentation

Model selection process for Regression: For the Regression analysis we need a type of predictive modelling which investigates the relationship between a dependent (target) and independent variable(s) (predictor). This technique is used for forecasting, time series modelling and finding the cause-effect relationship between the variables.

Evaluating the model accuracy is an essential part of the process in evaluating the performance of machine learning models to describe how well the model is performing in its predictions. The basic concept of accuracy evaluation is to compare the original target with the predicted one according to certain metrics. We use different models and interpret their values. We start by using linear regression in order to model the relationship between the features and the target variable. Second, we use Lasso regression as a type of linear regression that uses shrinkage. Shrinkage is where data values are shrunk towards a central point, like the mean. Like Ridge regression, Lasso is well-suited for models showing high levels of multicollinearity or when you want to automate certain parts of model selection, like variable selection/parameter elimination. XGBoost (Extreme Gradient Boosting) is used as an optimized distributed gradient boosting library and for supervised Machine Learning problems. XGBoost belongs to a family of boosting algorithms that convert weak learners into strong learners.

Regression

```
Text(0.5, 1.08, 'Regression Features')
```

Classification

Exploratory data analysis

The first impression of the data is that all attributes are important. It's the best to be rated around 7 - 8 to get a match.

We can also see that the data is very unbalanced, where only 1/5 of the dataset is marked as match while the other 4/5 is no match. This may influence the model.

We do a train/test split with 80/20% of the data.

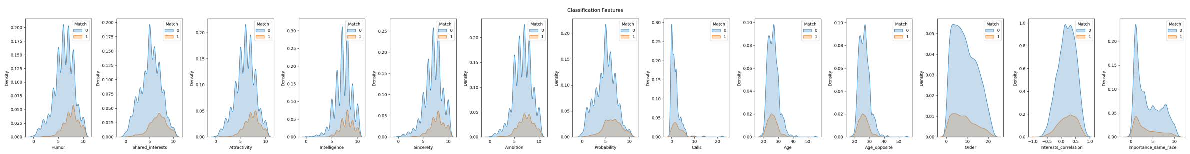
```
alt.VConcatChart(...)
```

When we are looking at those charts we search for differences between the match and no match results.

For **humor**, **shared interests**, **attractivity** and **probability** we can see again that the charts for **match** start to raise at around five, with a peak at 7-8. The importance for **same race** falls stronger for a match than for no match.

For the other attributes, the charts look pretty similar just with lower amplitudes.

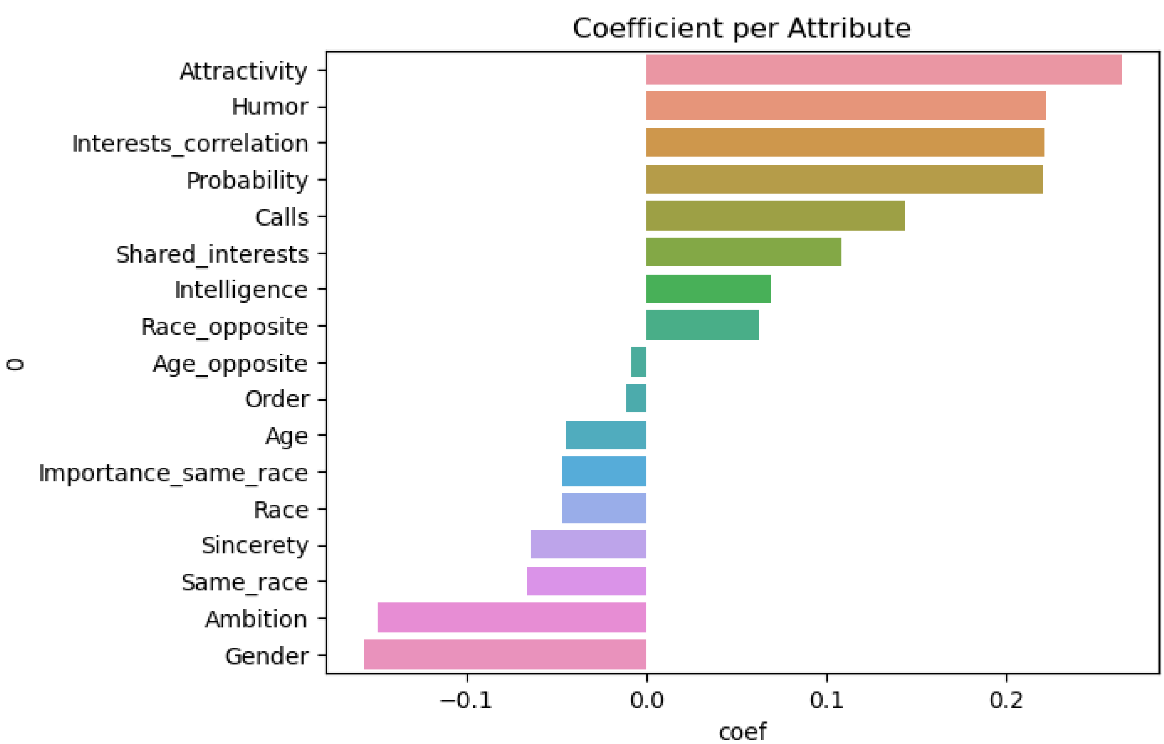
We can conclude that the mentioned attributes are probably important for the model in contrast to the others.



Model selection

The most important coefficients for a positive correlation for our model is **Attractivity**, followed by **Humor**, **Interests_correlation** and **Probability**. We also have strong negative correlations with **Gender** and **Ambition**.

```
[Text(0.5, 1.0, 'Coefficient per Attribute')]
```



Based on the metrics our model performs poor, predicting a lot of no matches. The R1 score for a match is very low (0.24). There are only 33 cases where we do a correct prediction of the match outcome.

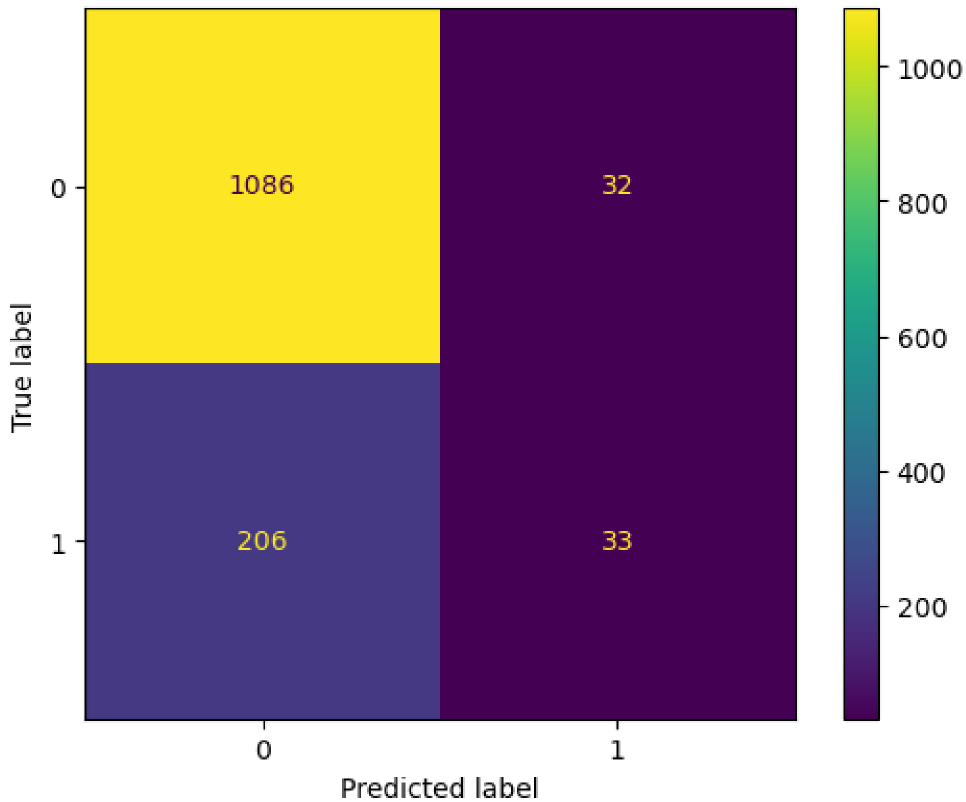
This may be based on the origin data where we have a lot more no-match entries than matches.

The **F1-Score** is calculated by: 2 * Precision * Recall / Precision + Recall. It takes the harmonic mean of precision and recall into account when optimizing the model. Values closer to 1 indicate a better performance.

The **Accuracy** is calculated by: Number of correct predictions / Total number of predictions. It's the portion of correct predictions.

Because our model does a lot of correct (true negative) predictions, the Accuracy score is high.

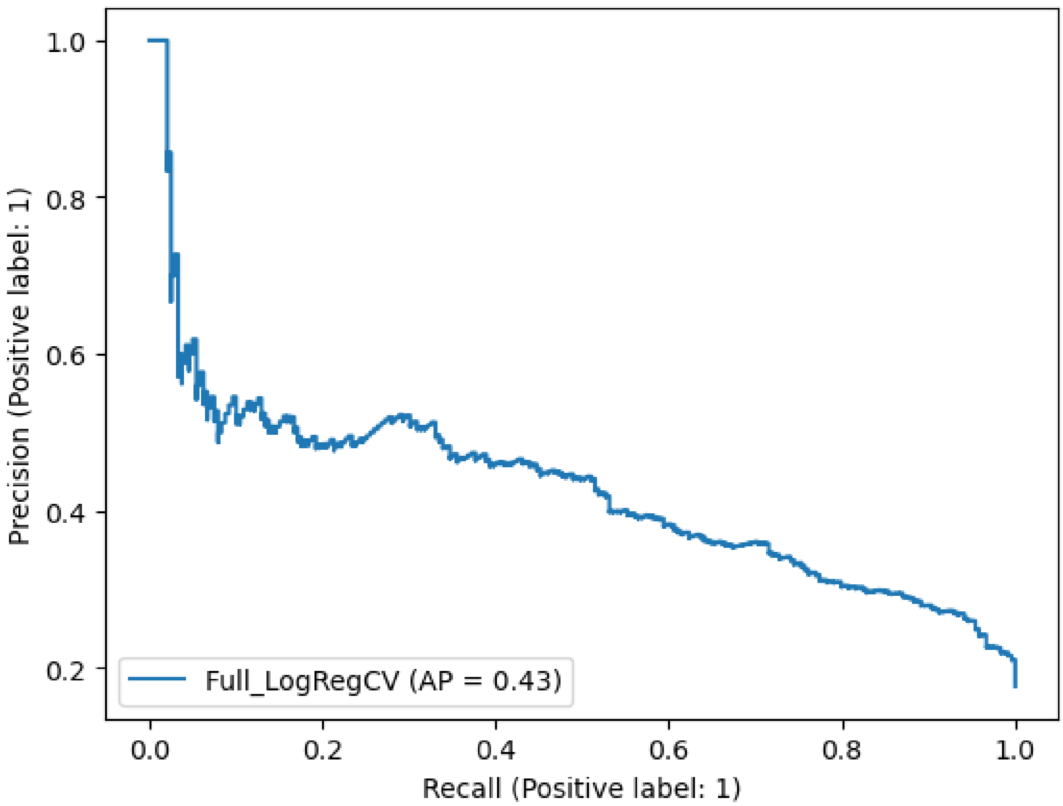
	precision	recall	f1-score	support
No match	0.84	0.97	0.90	1118
Match	0.51	0.14	0.22	239
accuracy			0.82	1357
macro avg	0.67	0.55	0.56	1357
weighted avg	0.78	0.82	0.78	1357



	Full_LogRegCV
Accuracy	0.824613
Precision	0.674125
Recall	0.554726
F1-score	0.559175

The **Precision-Recall** curve summarizes the trade-off between the **true positive rate (Recall)** and the **positive predictive value (Precision)**.

```
<sklearn.metrics._plot.precision_recall_curve.PrecisionRecallDisplay at 0x18bac4ded90>
```

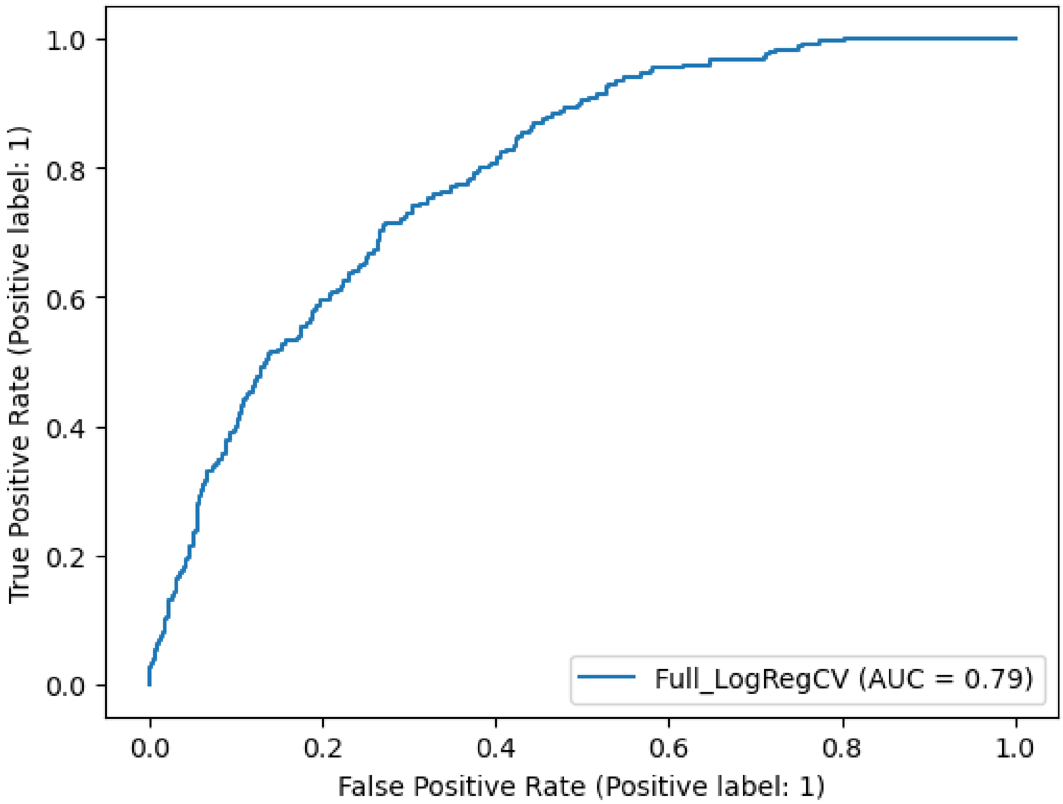


The **Receiver Operating Characteristic (ROC) Curve** summarizes the trade-off between the **true positive rate** and **false positive rate**.

ROC curves are appropriate when the observations are balanced between each class, whereas precision-recall curves are appropriate for imbalanced datasets.

Therefore the ROC curve looks good, although our model is in fact bad. This is caused by the high **true negative** rate in our model that is taken into account in the ROC but not in the Precision-Recall metric.

```
The AUC score is: 0.7909559060186676
```



We can try to even the numbers and train the model again.

```
alt.Chart(...)
```

Result

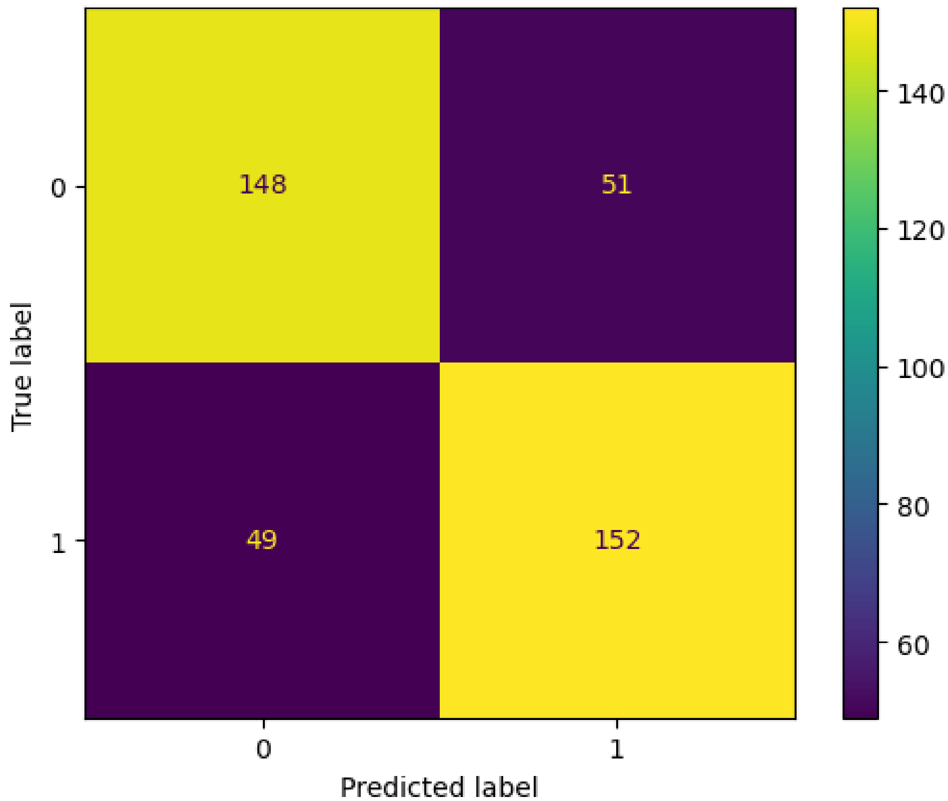
Our model looks a lot better now with a **F1 score** of 0.75. We do a correct estimation of a match in 152 of the cases and a correct estimation of no-match in 148 so in 3/4 of all cases we are correct and the **Accuracy** is at 75%.

Based on the dating behaviour it may be better to maximise the precision of match (have a lot of dates but less matches) or recall (have less dates but more matches). Because a match is still very personal, it is probably better to tune for precision.

```
LogisticRegressionCV(max_iter=200)
```

```
              precision    recall  f1-score   support
```

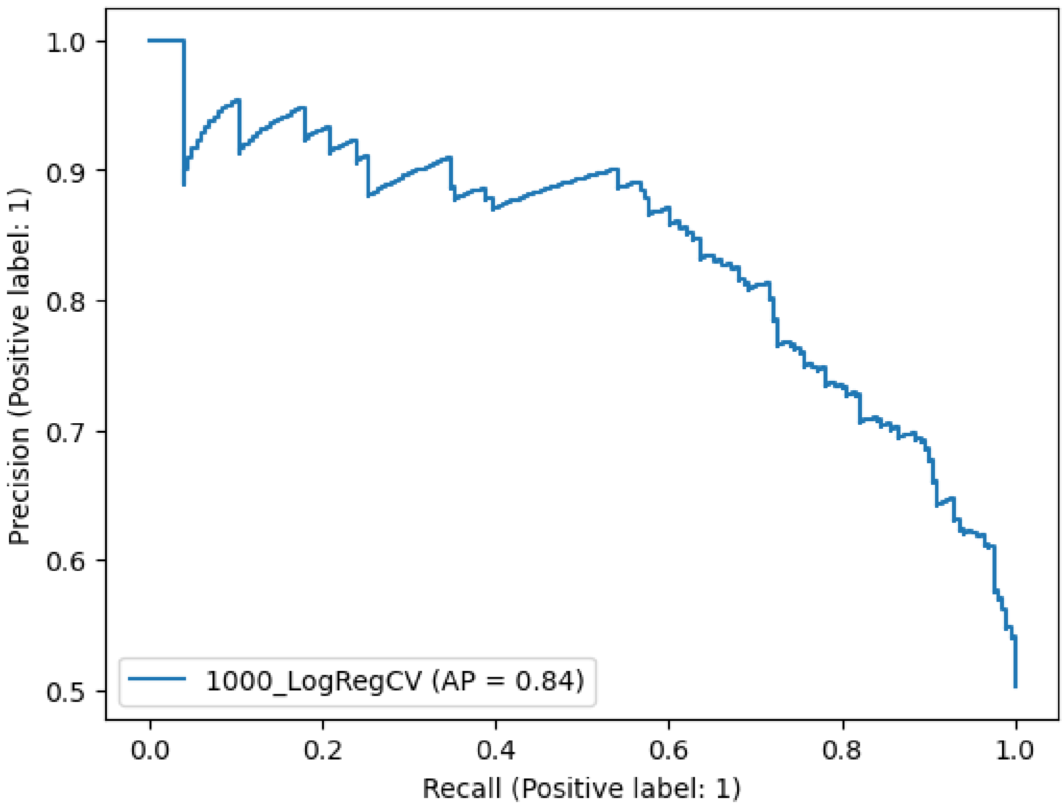
```
     No match       0.75      0.74      0.75       199
        Match       0.75      0.76      0.75       201

     accuracy                           0.75       400
    macro avg       0.75      0.75      0.75       400
 weighted avg       0.75      0.75      0.75       400
```



	1000_LogRegCV
Accuracy	0.750000
Precision	0.750019
Recall	0.749969
F1-score	0.749975

While the **AUC Score** stays roughly the same, the **Precision-Recall** curve looks a lot better now.

```
<sklearn.metrics._plot.precision_recall_curve.PrecisionRecallDisplay at 0x18bab259460>
```




```
The AUC score is: 0.8448961224030601
```

Tuning

We could tune the model very hard, so we *could* predict 22 partners and 21 of them would be a real match (in theory). See appendix.

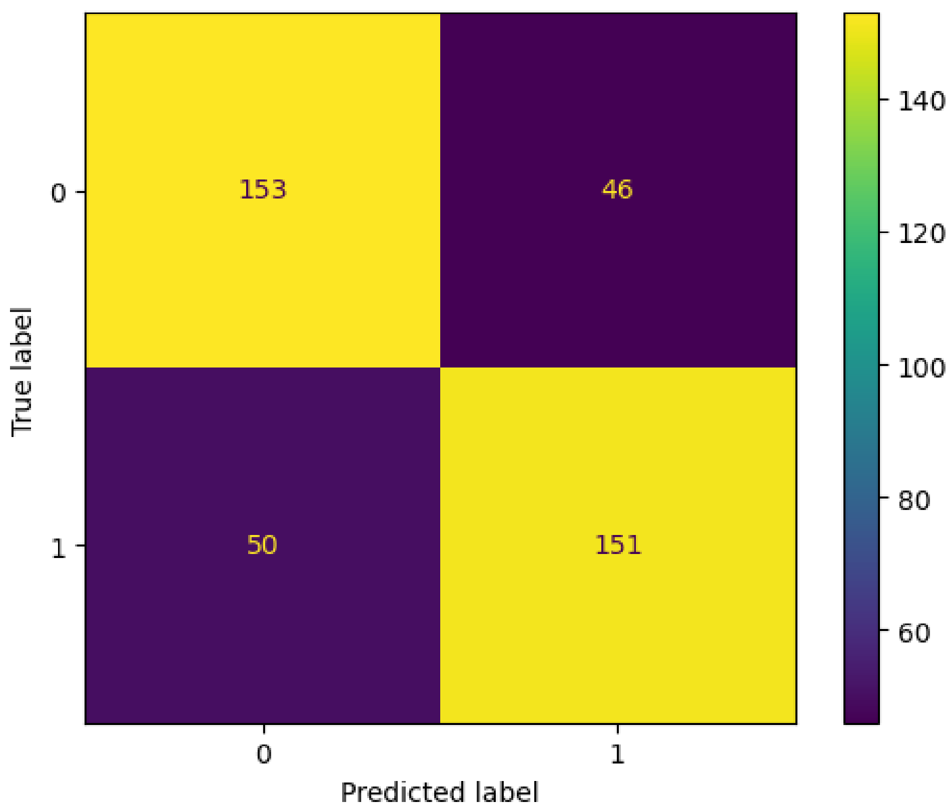
In this case we tune for optimal f1 score with a GridSearch.

```
Fitting 5 folds for each of 364 candidates, totalling 1820 fits
```

```
GridSearchCV(cv=StratifiedKFold(n_splits=5, random_state=0, shuffle=True),
             estimator=LogisticRegression(max_iter=200), n_jobs=-1,
             param_grid=[{'max_iter': [10000], 'penalty': ['none'],
                          'solver': ['lbfgs', 'newton-cg', 'sag', 'saga']},
                         {'C': array([1.00000000e-04, 2.63665090e-04, 6.95192796e-04, 1.83298
       4.83293024e-03, 1.27427499e-02, 3.35981829e-02, 8.85866790e-...
```

```
4.83293024e-03, 1.27427499e-02, 3.35981829e-02, 8.85866790e-02,
2.33572147e-01, 6.15848211e-01, 1.62377674e+00, 4.28133240e+00,
1.12883789e+01, 2.97635144e+01, 7.84759970e+01, 2.06913808e+02,
5.45559478e+02, 1.43844989e+03, 3.79269019e+03, 1.00000000e+04]),
                    'l1_ratio': array([0. , 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9
                    'max_iter': [10000], 'penalty': ['elasticnet'],
                    'solver': ['saga']}],
       scoring='accuracy', verbose=1)
```

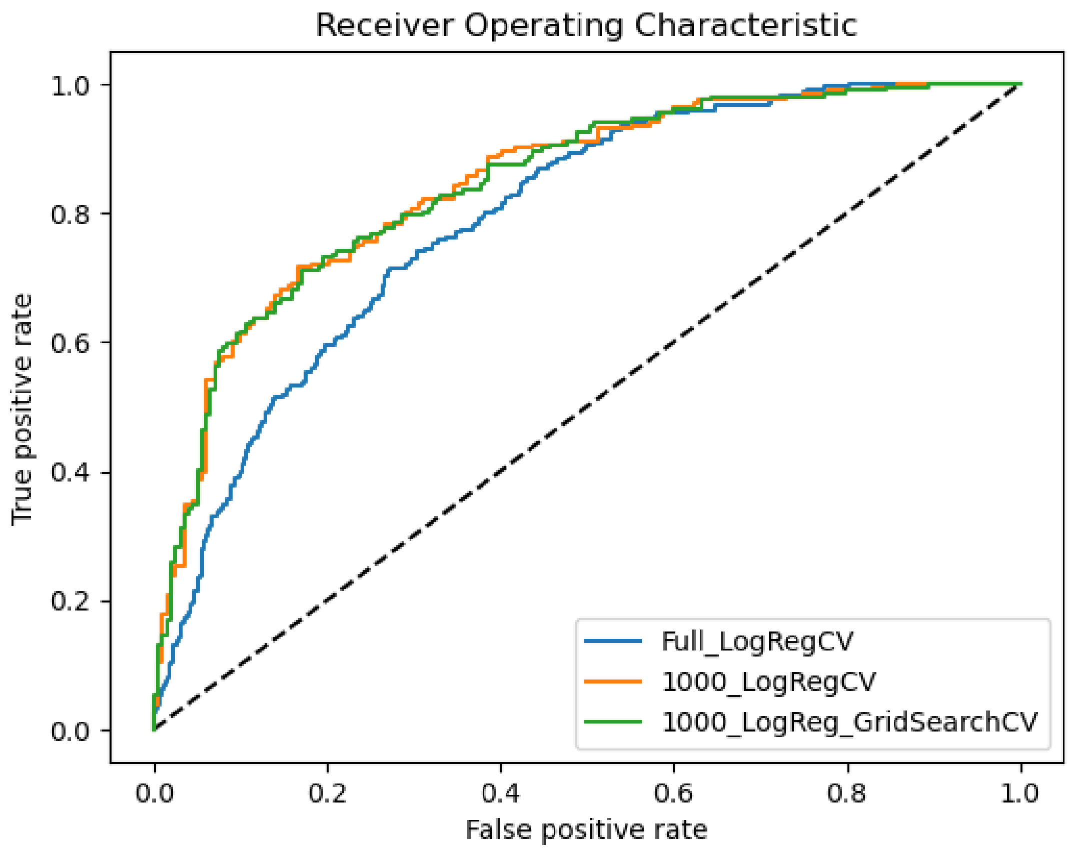
```
              precision    recall  f1-score   support

    No match       0.75      0.77      0.76       199
       Match       0.77      0.75      0.76       201

    accuracy                           0.76       400
   macro avg       0.76      0.76      0.76       400
weighted avg       0.76      0.76      0.76       400
```



Comparing the models, there is no big difference between the first model with a dataset of 2000 observations and the tuned one but balancing the dataset had a great effect on the model.

```
alt.VConcatChart(...)
```

```
The AUC score is: 0.8440961024025601
```



Discussion + Conclusion

With the data from this survey and our used statistical methods and classification, we could answer the following research questions:

- **What are the most effective personal characteristics to achieve a match in opposite sex speed dating?**

- According to our model parameters the most important features for a positive correlation are: **attractivity**, **same interests**, **humor** and that the other person also shows some **interest**.On the other hand there is a negative correlation for beeing the same gender and race and being (maybe too) ambitious and sincere.

Null hypothesis:

- **There is no affection of having specific characteristics regarding match selection of the survey participants**
  - This does not hold true, the characteristics are described above.
- **There is no correlation between shared interests, attributes and getting a match**
  - This does not hold true, the characteristics are described above.

Hypotheses:

- **Survey participants who both have the specific characteristics same race and opposite gender tend to achieve more matches**
  - We didn't investigate that in detail, but we saw a rather negative correlation between same race and match.
- **Survey participants with a higher income tend to achieve more matches than survey participants with a lower income**
  - We didn't investigate that in detail, as the income wasn't an important feature for our model.
- **Achieving matches because of having the same specific characteristics occur in both sexes**
  - Yes, this hypotheses is true.
- **Three weeks after the event, males called women more often**
  - Yes, by the factor of four. See appendix.

With our report, we contribute to a better understanding of the topic of speed dating and the preferences of the participants. Our paper serves as an important starting point in understanding the preferences underlying the search for a partner. Prior work has shown how to achieve matches, but in this report we compare these needed features and give an example which attributes a speed dating participant need to have in order to achieve matches and likes. In this report, we use an explorative data analysis approach that allows us to directly observe individual decisions.

There are a number of ways that our work may be improved. Due to the limitation of the data collection method - a local survey in only one country, we have a very specific distribution of races throughout the speed dating participants. Also, in terms of the validity of our dataset, gender politics have changed since 2008, and we have largely ignored gender diversity and focused only on men and women, altough those two genders don't really show a significant difference in the data. Most notably, a similar methodology could be employed on a newer set of data, because our data set is more than 10 years old.

Appendix

Data Dictionary

Descriptive terms for our used variables

Name	Description	Descriptive term
calls	Event of a participant conducting a "you_call" or "them_cal" with the other party	Calls of participants
attr	Rating of the attribute for this person from 1 - 10.	Attractivity of speed dating participant
sinc	Rating of the attribute for this person from 1 - 10.	Sincerety of speed dating participant
intel	Rating of the attribute for this person from 1 - 10.	Intelligence of speed dating participant
fun	Rating of the attribute for this person from 1 - 10.	Humor of speed dating participant

Name	Description	Descriptive term
amb	Rating of the attribute for this person from 1 - 10.	Ambition of speed dating participant
shar	Rating of the attribute for this person from 1 - 10.	Shared Interests/Hobbies of the speed dating participant to the other party
like	Overall, how much do oyu like this person. 1 (don't like at all) to 10 (like a lot)	Strength of like of speed dating participant to the other party
prob	How probable do you think it is that this person will say 'yes' for you? 1 (not probable) to 10 (extemely probable)	Probability of speed dating participant to like the other party

Name	Description	Descriptive term
met	Have you met this person before? (1 = yes, 2 = no)	Meeting indicator of participants
gender	Gender of the person. Female = 0, Male = 1	Gender of speed dating participant
order	The number of date that night when met partner	Order of date of speed dating participant and the other party during event
match	1 = yes, 0 = no	Match of the speed dating participant and the other party

Name	Description	Descriptive term
int_corr	Correlation between participant's and partner's ratings of interests in Time 1	Correlation of the speed dating participant and the other party
samerace	Participant and the partner were the same race. 1 = yes, 0 = no	Indicates, if the speed dating participant and the other party have the same race
age	Age of the person	Age of speed dating participant
age_o	Age of partner	Age of other party
race	Race of the attendee1 = Black/African American2 = European/Caucasian-American3 = Latino/Hispanic American4 = Asian/Pacific Islander/Asian-American5 = Native American6 = Other	Race of speed dating participant

Name	Description	Descriptive term
race_o	Race of partner	Race of other party
imprace	How important is it that a person you date be of the same racial/ethic background? (1 - 10)	Importance of the other party having the same race as the speed dating participant
intel_o	Intelligent. Rating by partner the night of the event from 1 (awful) to 10 (great)	Intelligence of the other party
sinc_o	Sincere. Rating by partner the night of the event from 1 (awful) to 10 (great)	Sincerety of the other party
like_o	Overall, how much do oyu like this person. 1 (don't like at all) to 10 (like a lot)	Strength of like of to the other party
prob_o	How probable do you think it is that this person will say 'yes' for you? 1 (not probable) to 10 (extemely probable)	Probability of the other party to like speed dating participant

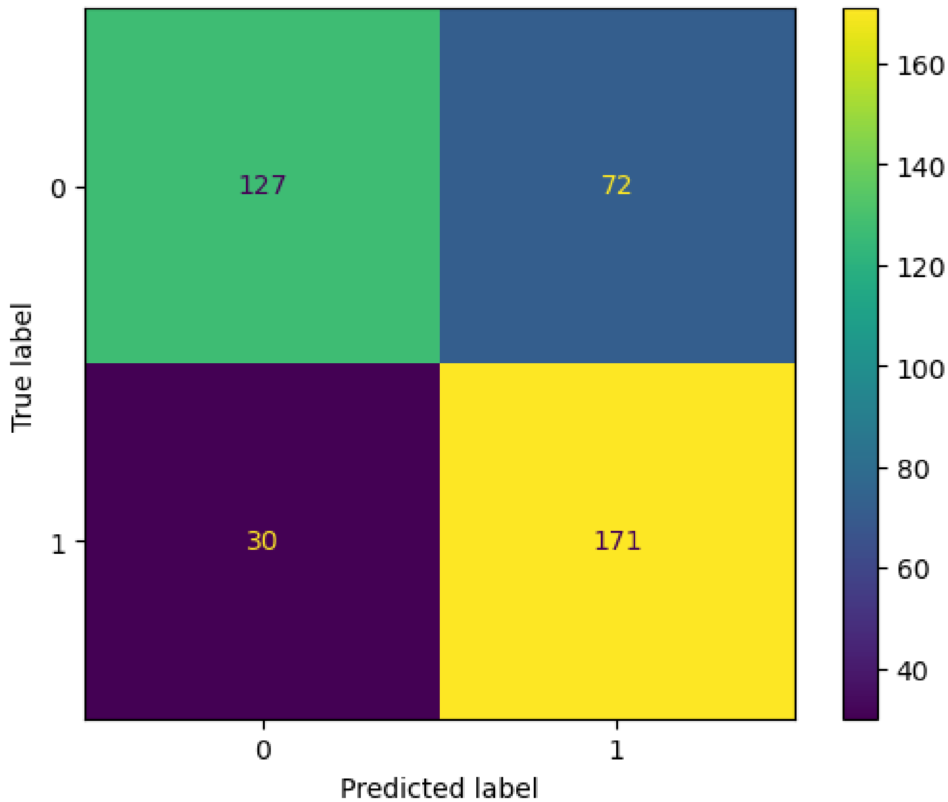
Name	Description	Descriptive term
fun_o	Fun. Rating by partner the night of the event from 1 (awful) to 10 (great)	Humor of the other party
satis_2	Generic Id	Generic Id
amb_o	Ambitious. Rating by partner the night of the event from 1 (awful) to 10 (great)	Ambition of the other party
shar_o	Shared Interests/Hobbies. Rating by partner the night of the event from 1 (awful) to 10 (great)	Shared Interests/Hobbies of the other party to speed dating participant
attr_o	Attractive. Rating by partner the night of the event from 1 (awful) to 10 (great)	Attractivity of the other party
met_o	Have you met this person before? (1 = yes, 2 = no)	Meeting indicator of the other party
exphappy	Overall, on a scale of 1-10, how happy do you expect to be with the people you meet during the speed-dating event?	Expected Happiness of meeting people

Name	Description	Descriptive term
pid	partner's iid number	partner's iid number

Tuning

The lower the thresholds, the more false positives we have.

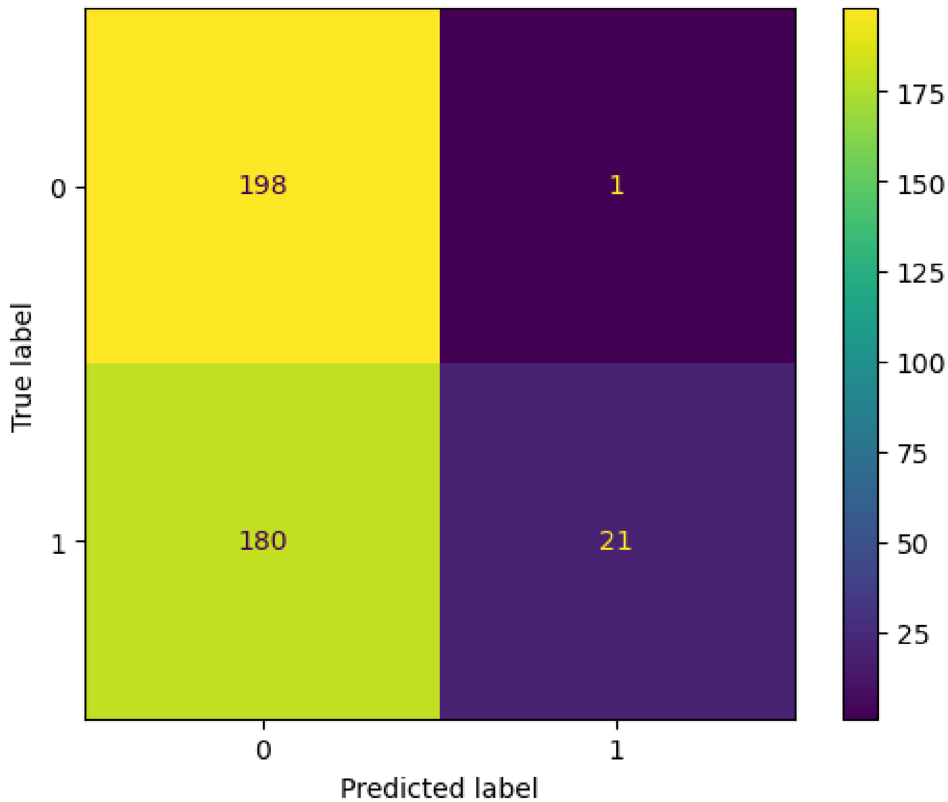
```
              precision    recall  f1-score   support

           0       0.81      0.64      0.71       199
           1       0.70      0.85      0.77       201

    accuracy                           0.74       400
   macro avg       0.76      0.74      0.74       400
weighted avg       0.76      0.74      0.74       400
```



The higher the thresholds, the more false negatives we have. If we want to tune it very hard, we can predict 22 partners and 21 of them would be a real match.

This would make sense if we want to make predictions for a person with whom the person should go on a date.

	precision	recall	f1-score	support
0	0.52	0.99	0.69	199
1	0.95	0.10	0.19	201
accuracy			0.55	400
macro avg	0.74	0.55	0.44	400
weighted avg	0.74	0.55	0.44	400



You call them call comparison of male and female

We can see that a lot more male (2.422) are calling female than the other way round (681). On the other hand, both sexes said that they have been called more often than there were actual calls (male 1.035/681 and female 2.866/2.422), maybe there is some bias about these numbers or the data is incomplete.

```
alt.HConcatChart(...)
```