Predicting the Origin of individuals from Genetic data

Team 17; https://github.com/Annilo/POrigGen

Ami Sild

1st year Masters student,
Data Science

Danat Yermakovich

3rd year PhD student,
Centre for Genomics,
Evolution and Medicine,
Institute of Genomics

Agnes Annilo

1st year Masters student,
Data Science

Grayson Felt

1st year Masters student,
Actuarial and Financial
Engineering

Predicting the Origin of individuals from Genetic data

Team 17; https://github.com/Annilo/POrigGen

Ami Sild

1st year Masters student,
Data Science

Danat Yermakovich

3rd year PhD student,
Centre for Genomics,
Evolution and Medicine,
Institute of Genomics

Agnes Annilo

1st year Masters student,
Data Science

Grayson Felt

1st year Masters student,
Actuarial and Financial
Engineering

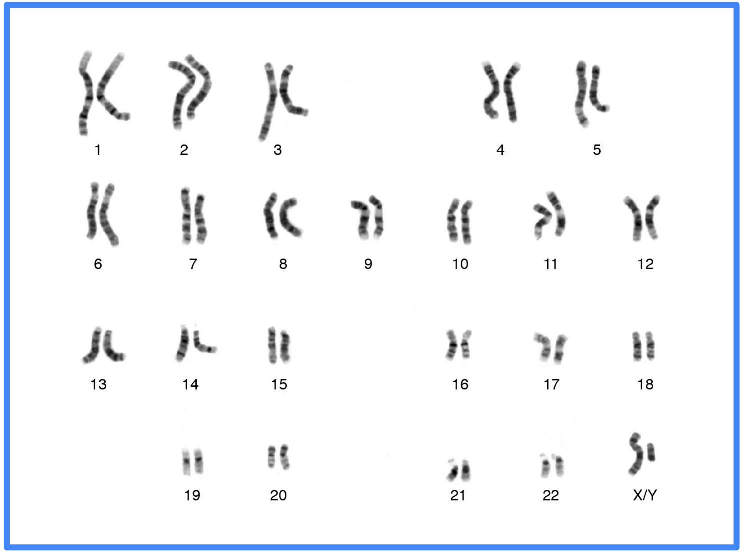
Predicting the Origin of individuals from Genetic data

Team 17; https://github.com/Annilo/POrigGen

Genealogical geographical origin

~

due to:
gradient changing
of
genetic population
structure
across world



Human genome

https://www.genome.gov/genetics-glossary/Karyotype

Approach

Predicting sample's population label from genetic data

80%
train

3200 samples (observations)
26 Pops across 5 SuperPop
50-150 samples per Pop

~10 millions genetic
variations i.e. features

2561 / 641

<table>
<tr><th>chr_pos_ref_alt</th><th>1:58771:T:C</th><th>1:183401:C:G</th><th>1:186291:G:A</th><th>1:281912:C:G</th></tr>
<tr><th>SampleID</th><th></th><th></th><th></th><th></th></tr>
<tr><td>HG00097</td><td>1|1</td><td>0|0</td><td>0|0</td><td>0|0</td></tr>
<tr><td>HG00099</td><td>0|0</td><td>0|0</td><td>1|0</td><td>0|0</td></tr>
<tr><td>HG00100</td><td>1|0</td><td>0|0</td><td>0|0</td><td>0|0</td></tr>
<tr><td>HG00101</td><td>1|0</td><td>0|0</td><td>0|0</td><td>1|0</td></tr>
<tr><td>HG00102</td><td>1|1</td><td>0|0</td><td>0|0</td><td>1|0</td></tr>
<tr><td>HG00103</td><td>0|0</td><td>0|0</td><td>0|0</td><td>0|0</td></tr>
<tr><td>HG00105</td><td>0|1</td><td>0|0</td><td>0|0</td><td>1|0</td></tr>
<tr><td>HG00106</td><td>0|1</td><td>0|1</td><td>0|0</td><td>0|0</td></tr>
<tr><td>HG00107</td><td>1|0</td><td>0|0</td><td>0|0</td><td>1|0</td></tr>
</table>

Approach

Predicting sample's population label from genetic data

3200 samples (observations)
26 Pops across 5 SuperPop
50-150 samples per Pop

~10 millions genetic variations i.e. features

80% train

Genetic feature preprocessing (MAF < 0.05, LD pruning)

70 000 features

PCA

Train and evaluate different models

Outcome

predicting a sample's population label (1000G)

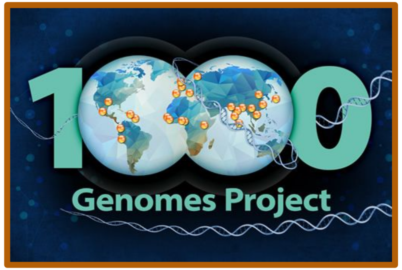
10 KK => 70K

2561 / 641

chr_pos_ref_alt	1:58771:T:C	1:183401:C:G	1:186291:G:A	1:281912:C:G
SampleID				
HG00097	1|1	0|0	0|0	0|0
HG00099	0|0	0|0	1|0	0|0
HG00100	1|0	0|0	0|0	0|0
HG00101	1|0	0|0	0|0	1|0
HG00102	1|1	0|0	0|0	1|0
HG00103	0|0	0|0	0|0	0|0
HG00105	0|1	0|0	0|0	1|0
HG00106	0|1	0|1	0|0	0|0
HG00107	1|0	0|0	0|0	1|0

Approach

Predicting sample's population label from genetic data



3200 samples (observations)
26 Pops across 5 SuperPop
50-150 samples per Pop

~10 millions genetic variations i.e. features

80% train

Genetic feature preprocessing (MAF < 0.05, LD pruning)

70 000 features

10 KK => 70K

PCA

Visualisation

Train and evaluate different models

CV-tuning of hyperparameters and N of PCs

Outcome

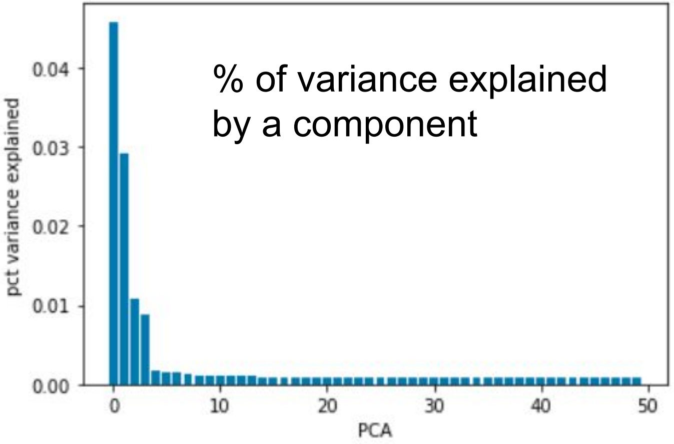
predicting a sample's population label (1000G)

2561 / 641

chr_pos_ref_alt	1:58771:T:C	1:183401:C:G	1:186291:G:A	1:281912:C:G
SampleID				
HG00097	1|1	0|0	0|0	0|0
HG00099	0|0	0|0	1|0	0|0
HG00100	1|0	0|0	0|0	0|0
HG00101	1|0	0|0	0|0	1|0
HG00102	1|1	0|0	0|0	1|0
HG00103	0|0	0|0	0|0	0|0
HG00105	0|1	0|0	0|0	1|0
HG00106	0|1	0|1	0|0	0|0
HG00107	1|0	0|0	0|0	1|0

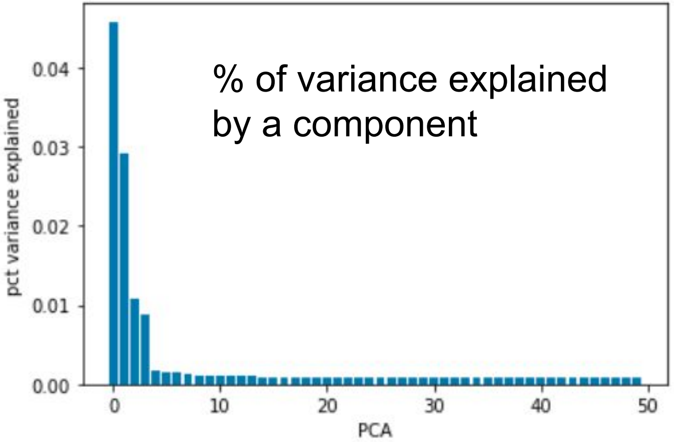
Results: Data PCs

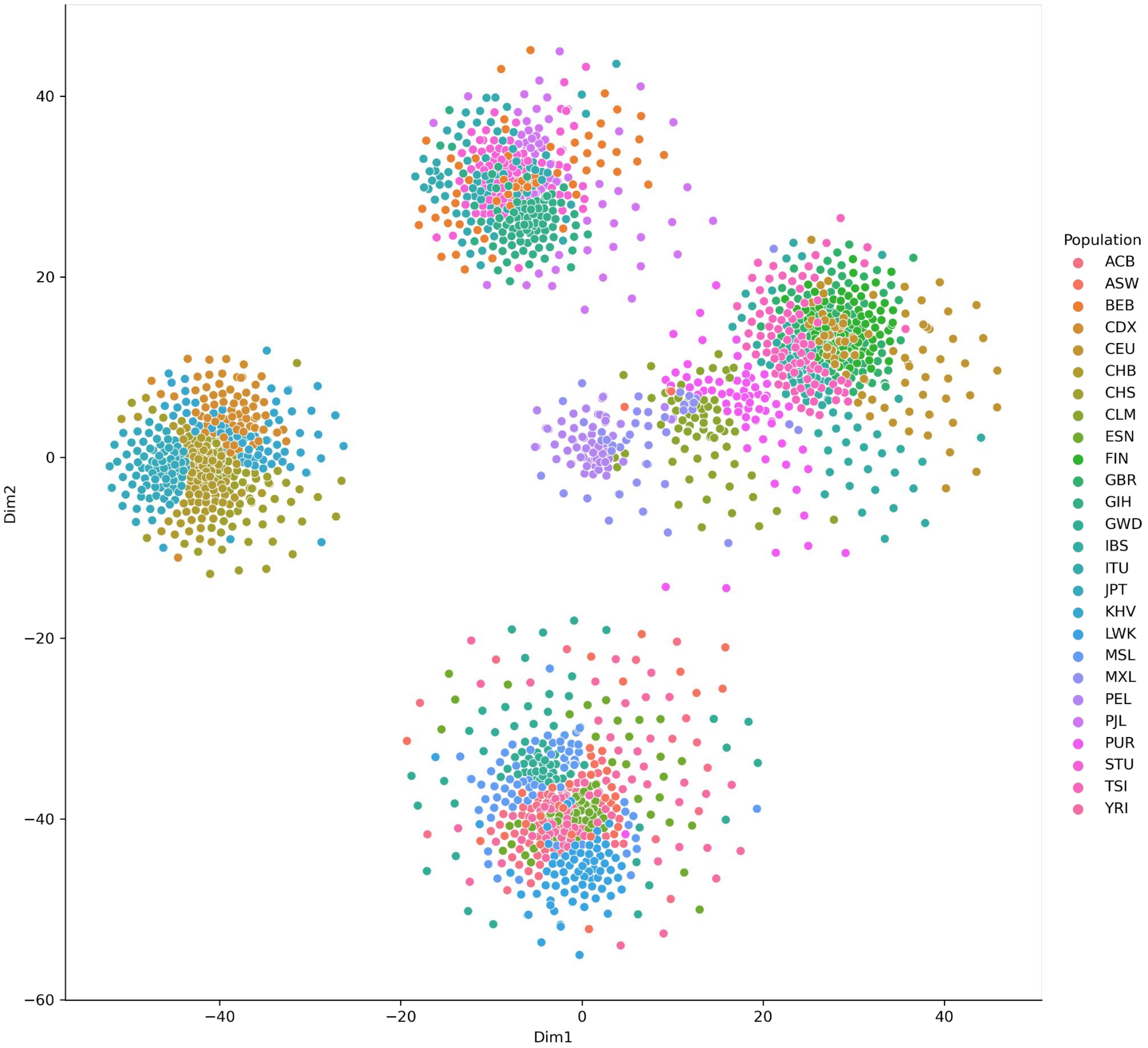
2561 train samples from 1000G:
all PCs



Results: Data tSNE

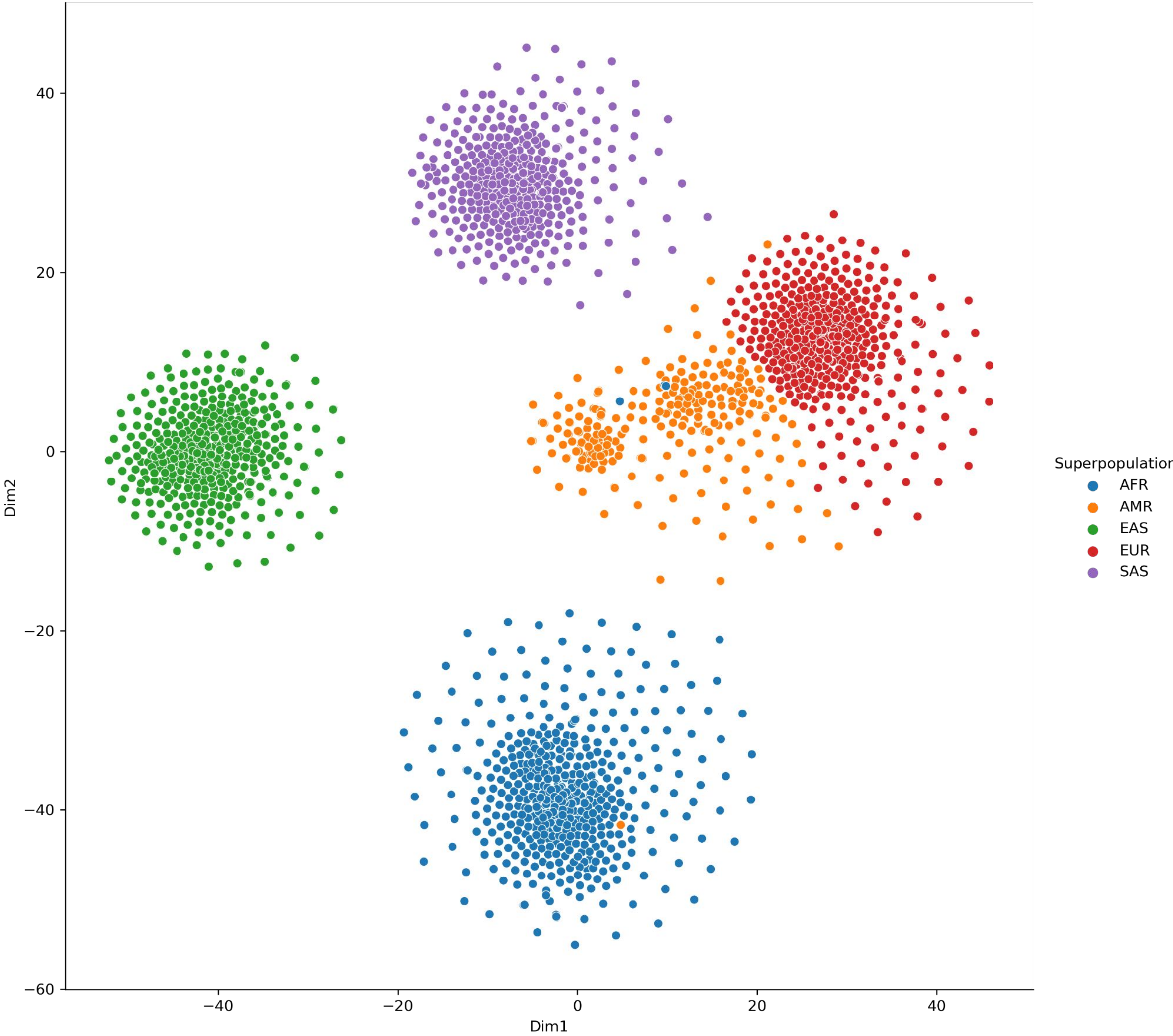
2561 train samples from 1000G: **all PCs**





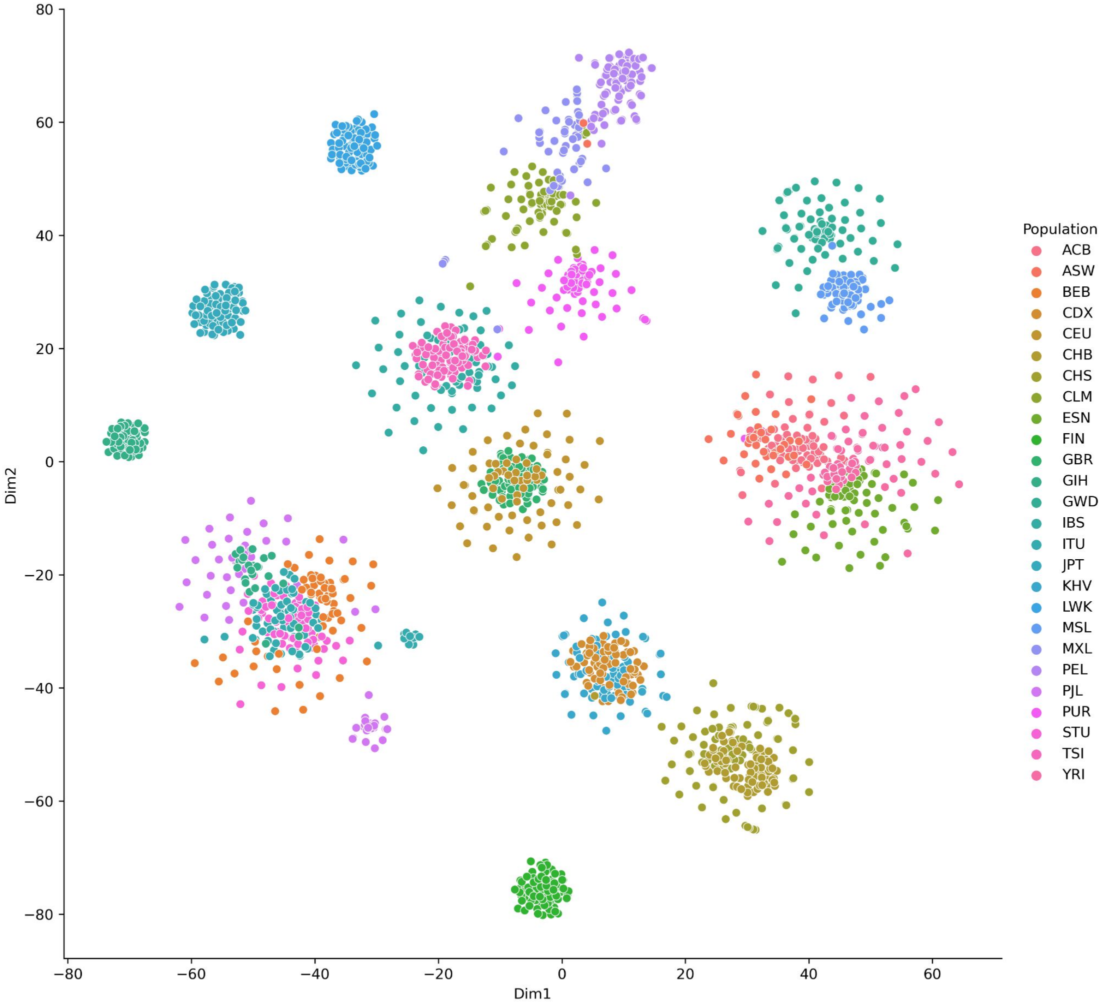
Results: Data tSNE

2561 train samples from 1000G:
all PCs



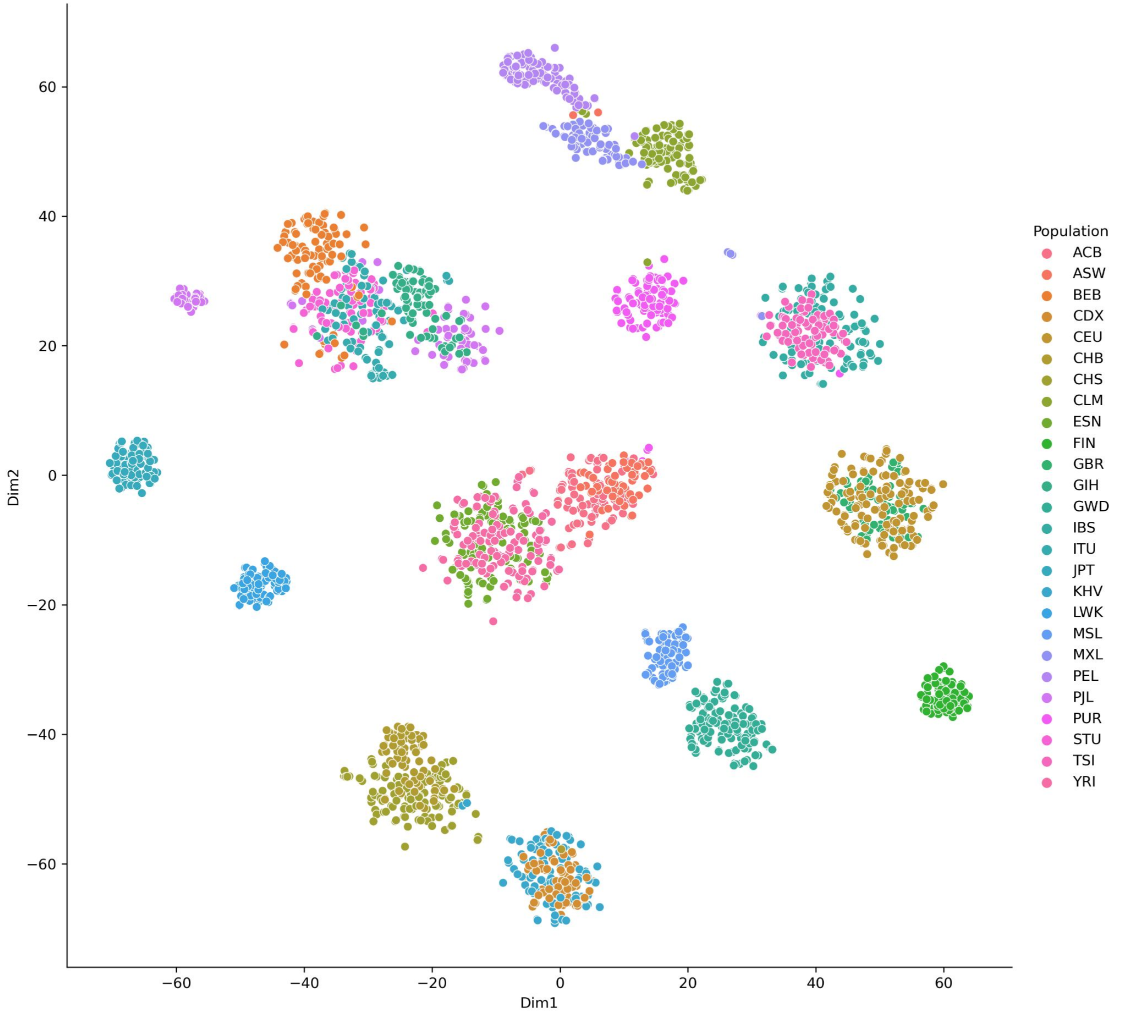
Results: Data tSNE

2561 train samples from 1000G:
50 PCs



Results: Data tSNE

2561 train samples from 1000G:
20 PCs

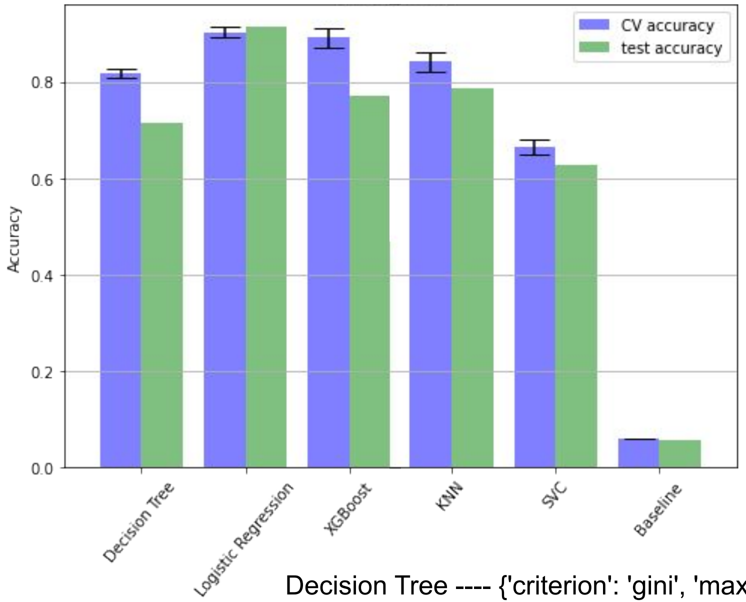


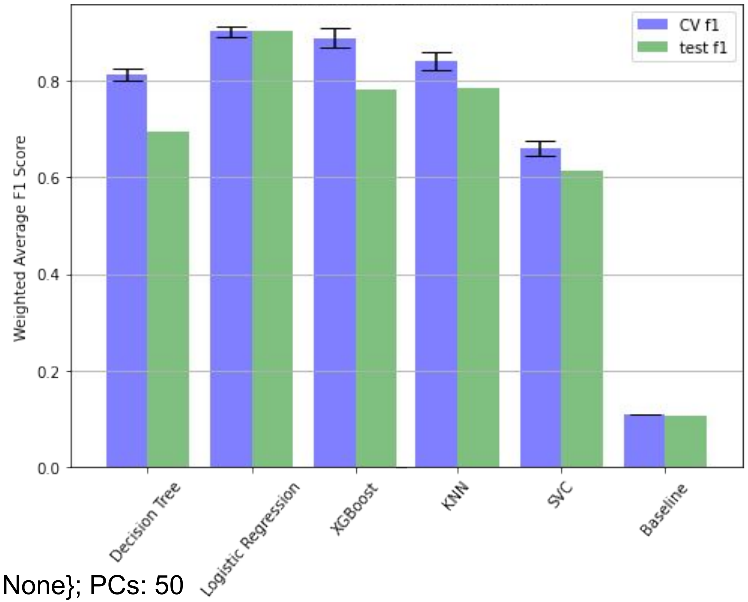
Results: Models

Accuracy

GridSearchCV, 4 folds

Weighted Average F1

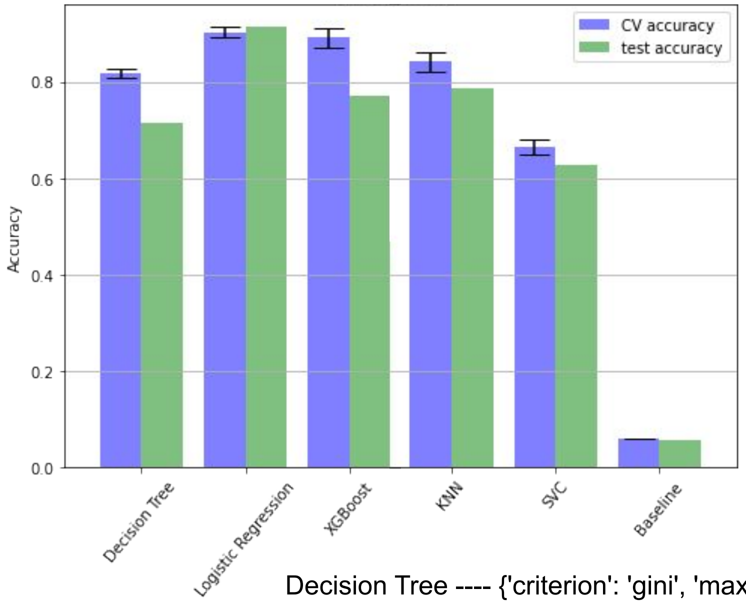


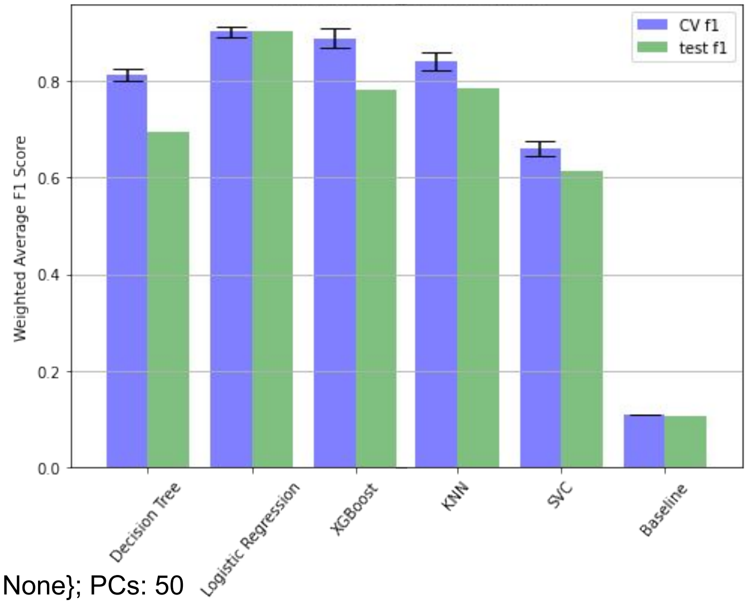


Decision Tree ---- {'criterion': 'gini', 'max_depth': None}; PCs: 50
Logistic Regression ---- {'penalty': 'l2', 'solver': 'saga'}; PCs: 1000
XGBoost ---- {'gamma': 0.5, 'max_depth': '100'}; PCs: 500
KNN ---- {'n_neighbors': 2, 'weights': 'distance'}; PCs: 50
SVC ---- {'kernel': 'poly'}; PCs: 5
Baseline ---- always predict largest class

Results: Models

Accuracy *GridSearchCV, 4 folds* Weighted Average F1





Decision Tree ---- {'criterion': 'gini', 'max_depth': None}; PCs: 50

Logistic Regression ---- {'penalty': 'l2', 'solver': 'saga'}; PCs: 1000

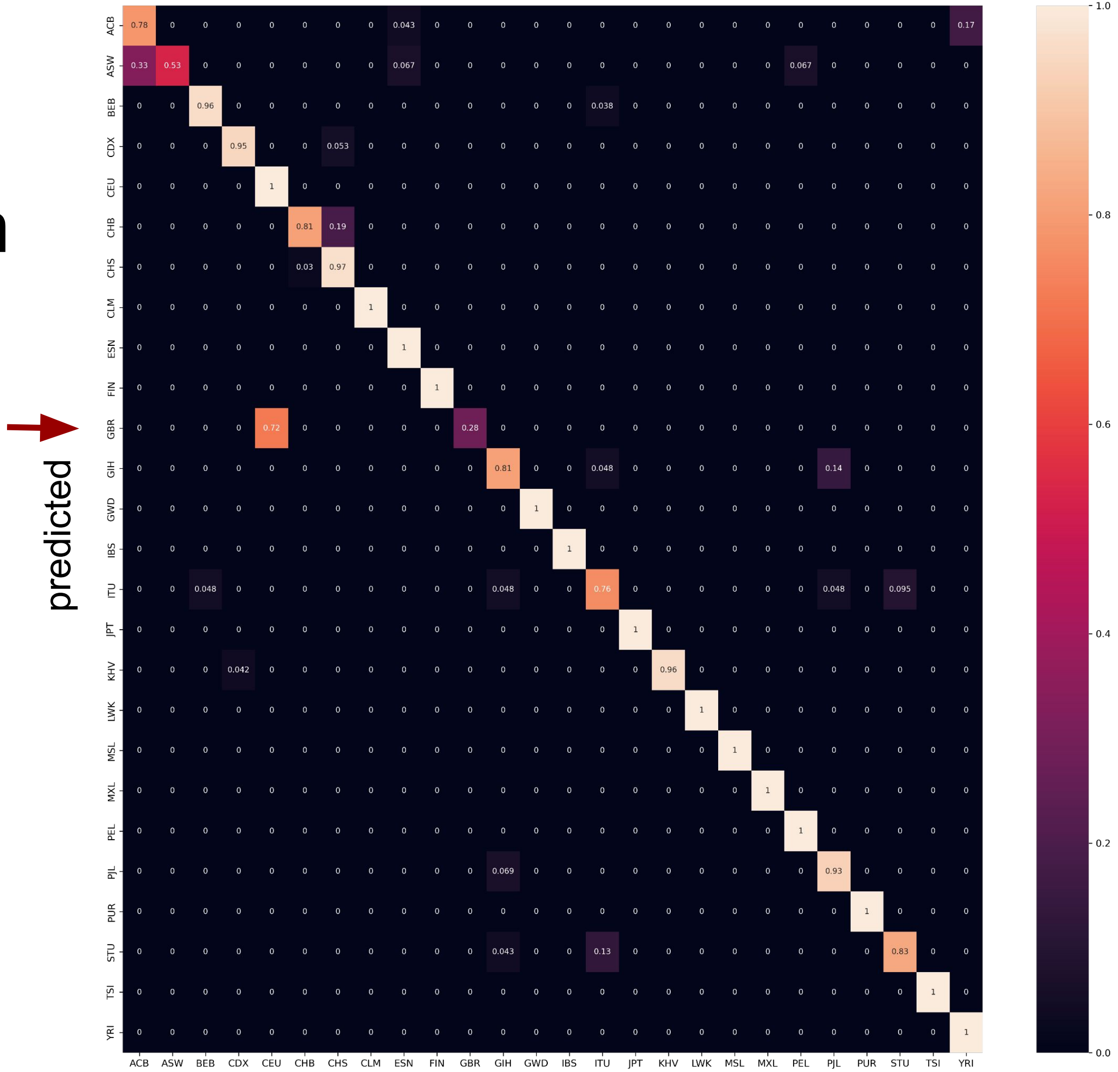
XGBoost ---- {'gamma': 0.5, 'max_depth': '100'}; PCs: 500

KNN ---- {'n_neighbors': 2, 'weights': 'distance'}; PCs: 50

SVC ---- {'kernel': 'poly'}; PCs: 5

Baseline ---- always predict largest class

Results:
Best LogRegression Confusion Matrix



Main Lessons

- Different stages of problems complexity have their own best types of models
- In multiclassification, primary efforts can be devoted to distinguishing the most similar classes
- Ensembles have potential in multiclassifaction
- Large datasets require a large RAM amount

https://github.com/Annilo/POrigGen

Main Lessons

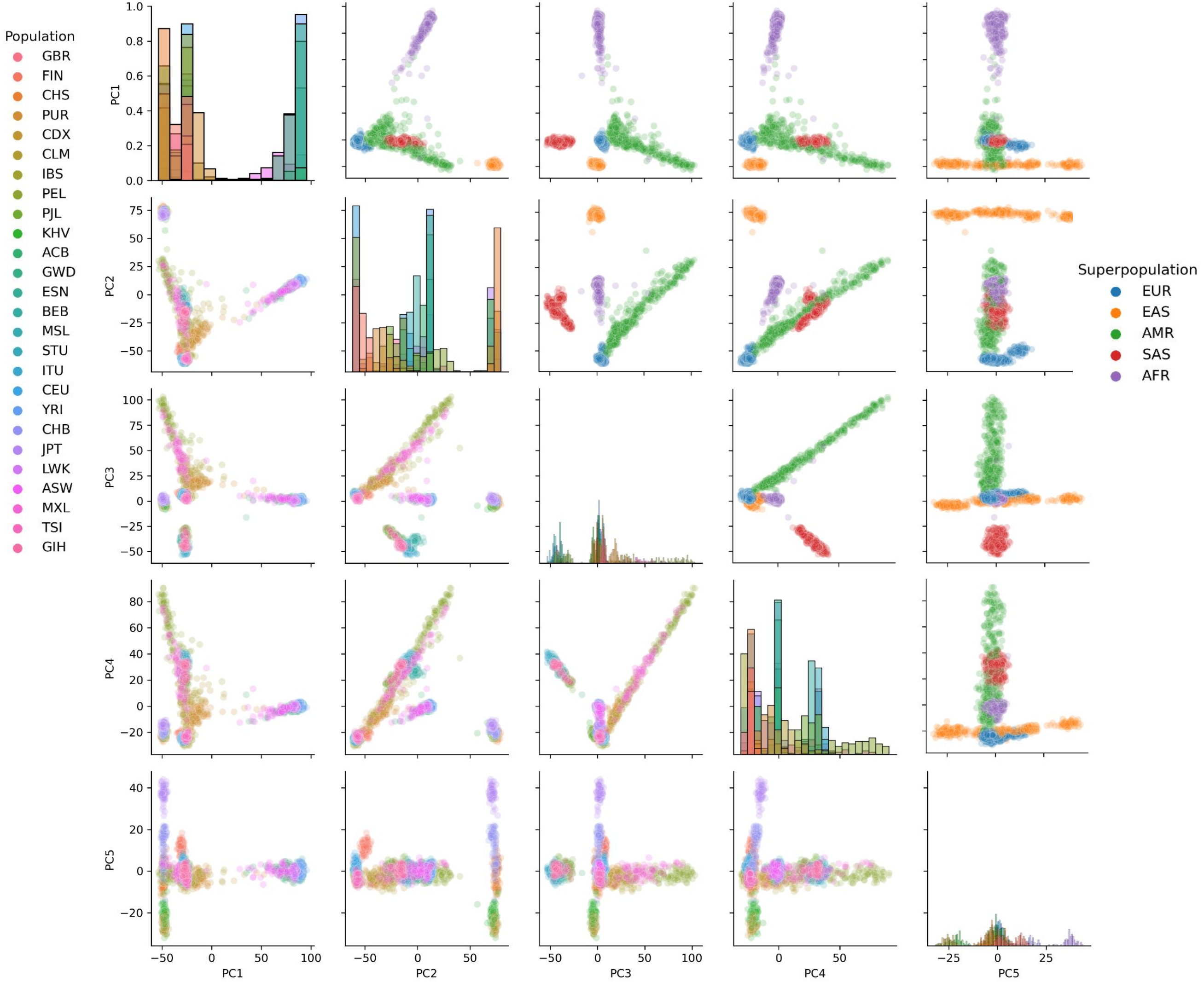
- Different stages of problems complexity have their own best types of models
- In multiclassification, primary efforts can be devoted to distinguishing the most similar classes
- Ensembles have potential in multiclassifaction
- Large datasets require a large RAM amount

Thank you for your attention!

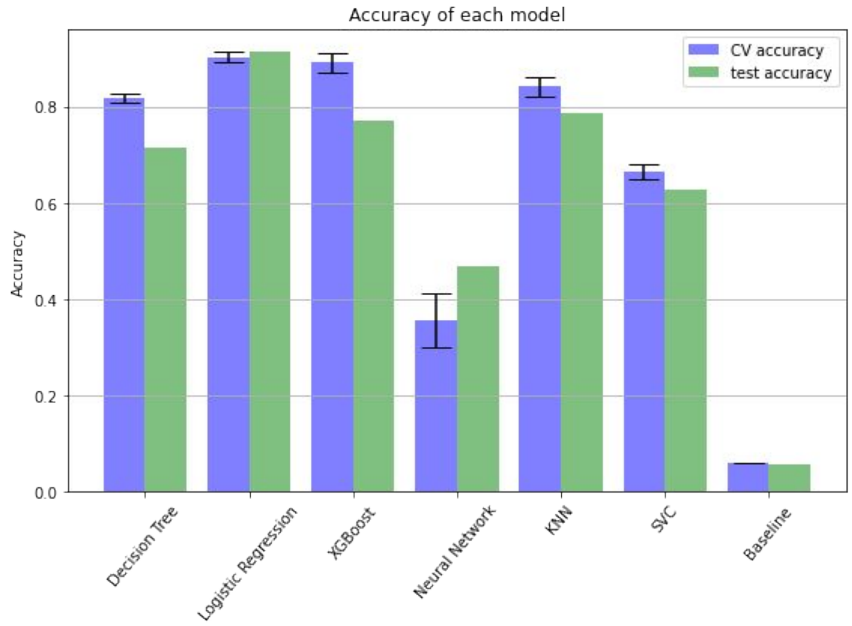
https://github.com/Annilo/POrigGen

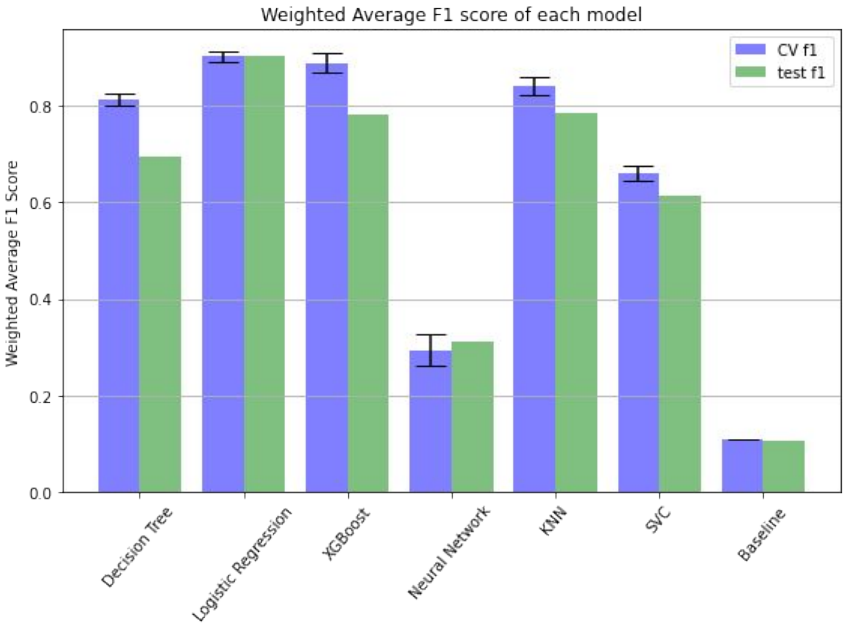
Results: Data PCs

2561 train samples from 1000G:
26 populations
5 Superpopulations



Results: Models





Decision Tree ---- {'criterion': 'gini', 'max_depth': None}; PCs: 50
Logistic Regression ---- {'penalty': 'l2', 'solver': 'saga'}; PCs: 1000
XGBoost ---- {'gamma': 0.5, 'max_depth': '100'}; PCs: 500
#Neural Network ---- {'activation': 'relu', 'solver': 'adam'}; PCs: 1000
KNN ---- {'n_neighbors': 2, 'weights': 'distance'}; PCs: 50
SVC ---- {'kernel': 'poly'}; PCs: 5
Baseline ---- always predict largest class

Results: Ensemble from RF, KNN, LogR