Task: Predicting Cognitive Performance Using Demographics

By: Anmin Yang & Shan Gao

- Cognitive training: How do we predict users' cognitive capacity to tailor the training program? Demographics?
- Classify participants' cognitive performance into 2 classes using demographic data
  - High cognitive performance
  - Low cognitive performance
- Classification problem with binary classes

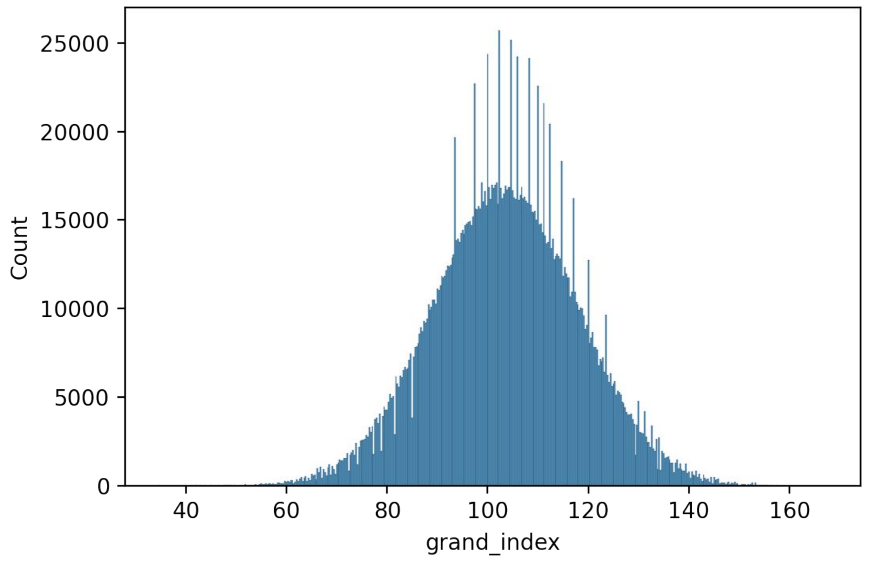
Data

Raw Data

(2302948, 11)

	user_id	age	gender	education_level	country	test_run_id	battery_id	specific_subtest_id	raw_score	time_of_day	grand_index
0	29	69.0	m	4.0	US	100605	50	29	14.0	22	87.413696
1	29	69.0	m	4.0	US	100605	50	45	28.0	22	87.413696
2	29	69.0	m	4.0	US	100605	50	43	6.0	22	87.413696
3	29	69.0	m	4.0	US	100605	50	44	9.0	22	87.413696
4	29	69.0	m	4.0	US	100605	50	39	53.0	22	87.413696
5	29	69.0	m	4.0	US	100605	50	40	53.0	22	87.413696

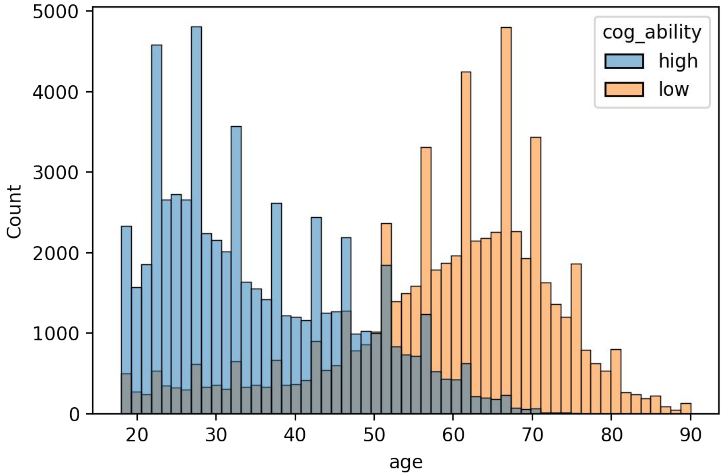
- All rows with missing values are dropped
- Duplicated rows are merged
- Redundant features are dropped
  - user_id
  - test_run_id
  - battery_id
  - specific_subtest_id
  - raw_score



Classification problem:

- Top 25% grand index: high cognitive ability
- Bottom 25% grand index : low cognitive ability
- 62679 observations per class

Data

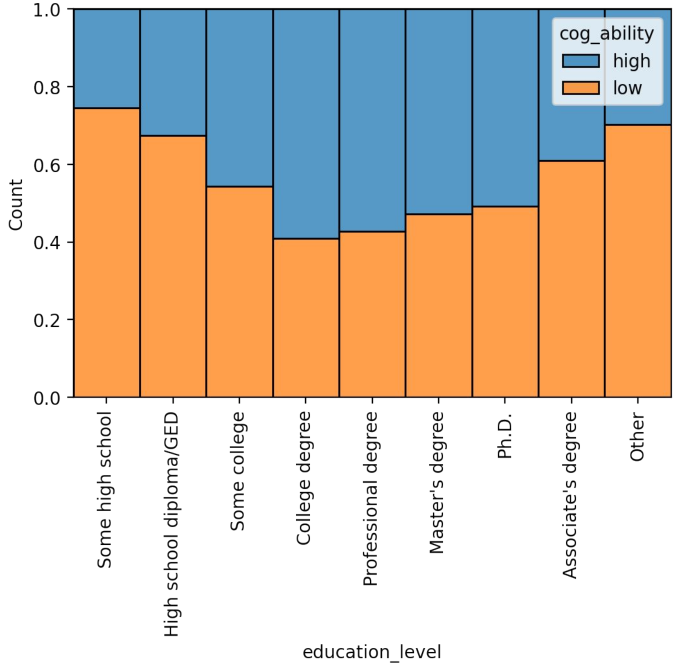


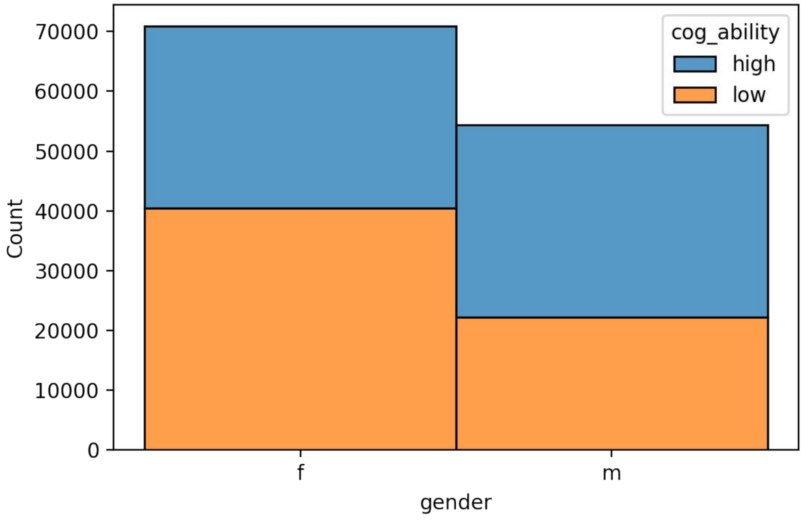
- 1 numeric feature
  - age
- 3 categorical features
  - education_level
  - gender
  - country

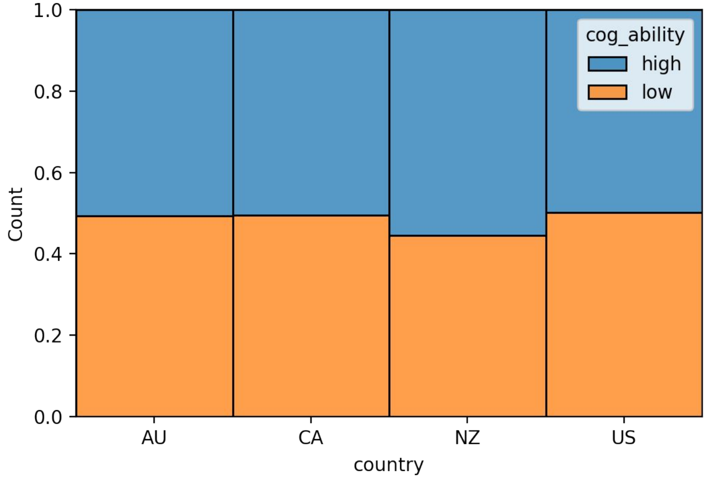
Cleaned DataFrame

(125358, 5)

	age	gender	education_level	country	cog_ability
0	79.0	f	4.0	US	low
1	51.0	m	6.0	US	low
2	33.0	f	99.0	US	low
3	76.0	m	4.0	US	low
4	54.0	m	6.0	US	low







Data

Detecting Multicollinearity with VIF

	feature	VIF
0	age	4.928579
1	gender	1.642837
2	education_level	2.463673
3	country	4.894018

Feature Engineering

- one-hot encoding for categorical features
- standardization of numeric feature (age) for logistic regression

Train-Test Split

- 80% as training set
- 20% as test set
- with *train_test_split* function

Classes are splitted equally

- Test Data
    - 12570 high cognitive performance
    - 12502 low cognitive performance

Logistic Regression

- Grid-search for hyperparameters (cv = 10)
  - parameters = [{'penalty':['l1','l2'], 'C':[0.1, 1, 10, 100, 1000]}]
- AUC as the score criteria

- Best AUC score: 0.9041449584055424
- Best parameter combination: {'C': 0.1, 'penalty': 'l1'}

Weights

female	male
0.25	0

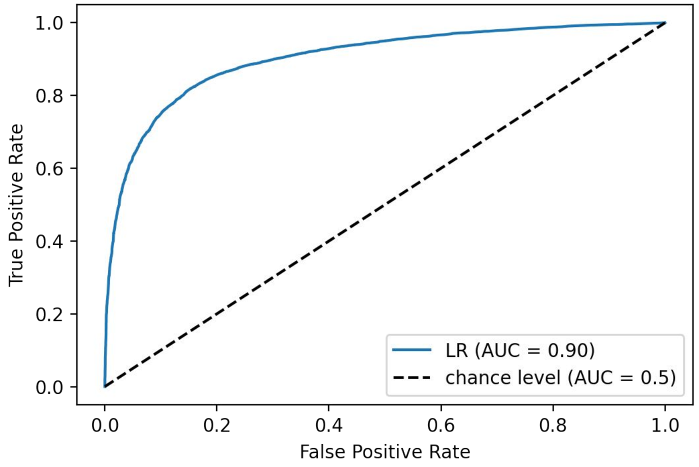
AU	CA	NZ	US
0	0.11	0	0.04

age
2.11

High : 0; Low: 1

Some high school	2.11
High school diploma/GED	1.19
Other	1.34
Some college	0.35
Associate's degree	0.34
College degree	-0.58
Professional degree	-0.77
Master's degree	-0.79
Ph.D.	-0.99

	precision	recall	f1-score	support
0	0.84	0.83	0.83	12502
1	0.83	0.84	0.83	12570
accuracy			0.83	25072
macro avg	0.83	0.83	0.83	25072
weighted avg	0.83	0.83	0.83	25072



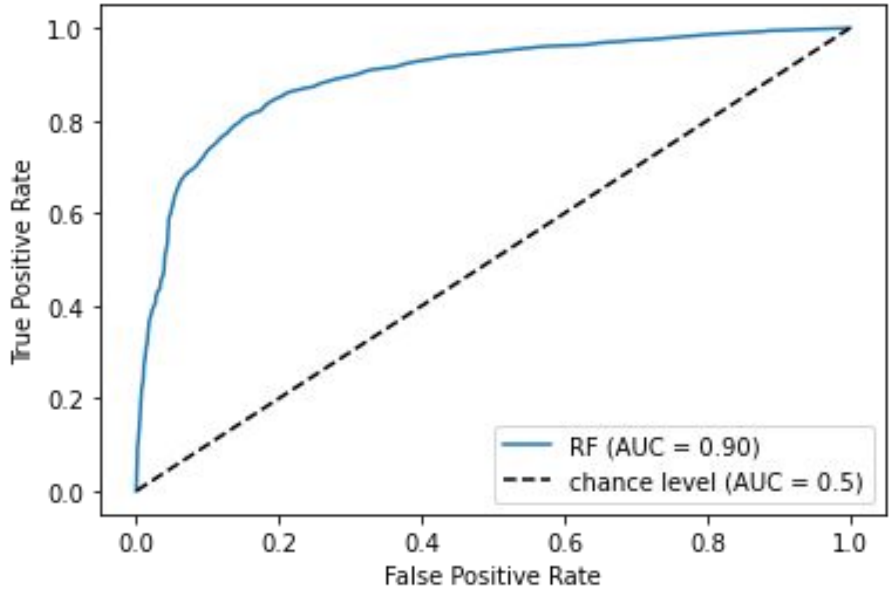
Failed Case

Unnamed: 0	78454
user_id	66091868
age	53.0
gender	f
education_level	5.0
country	CA
time_of_day	9
cog_ability	high

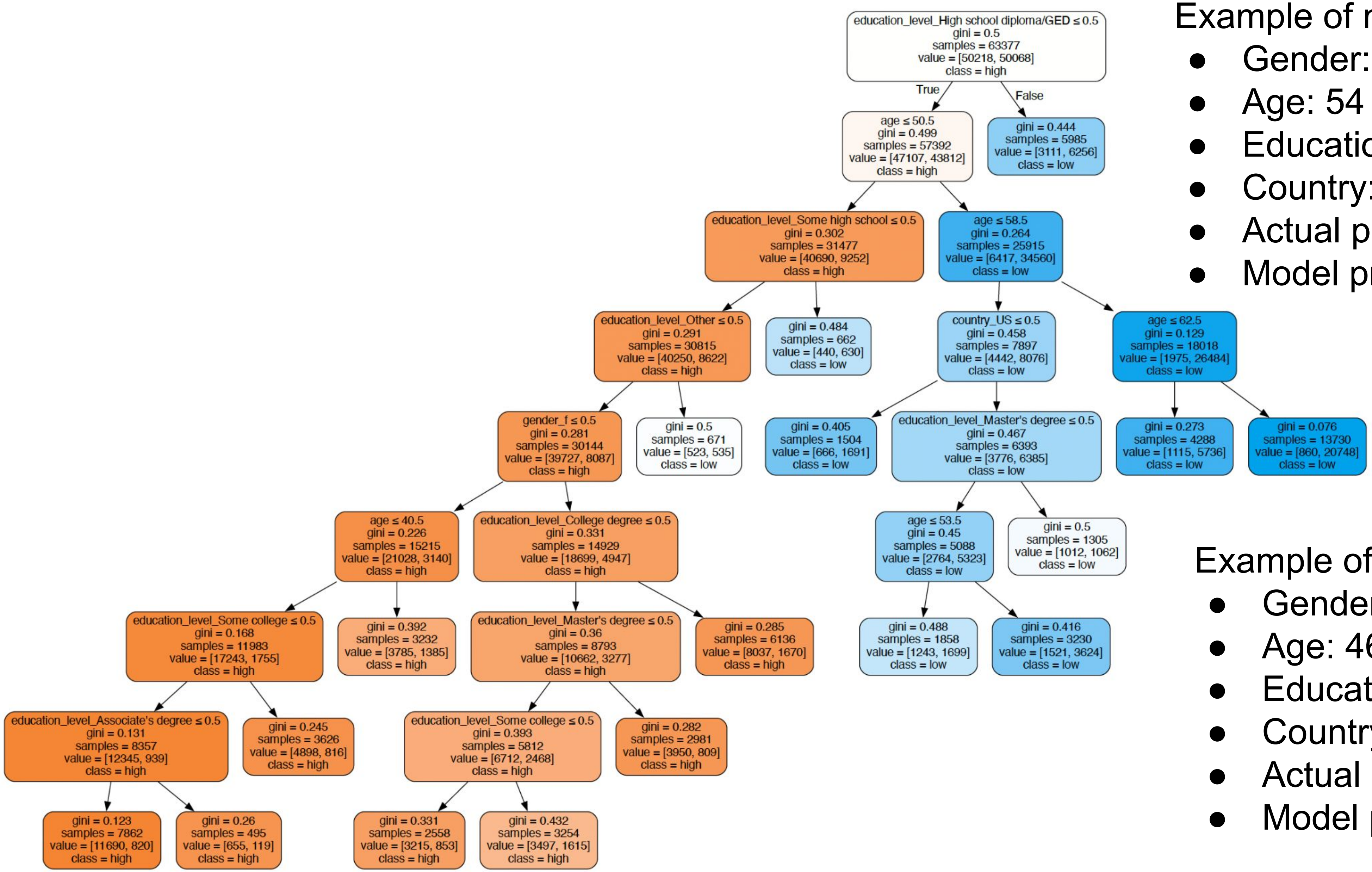
Random Forest – Model

- Numeric value transferred to original
  - scaler.inverse_transform()
- Parameter selection:
  - Grid search
    - 'min_samples_split': np.linspace(0.05, 0.6, num=10)
    - 'max_leaf_nodes': np.arange(2,20,3)}]
  - Manual
    - 'max_depth': None, 5, 3
    - 'min_sample_leaf': 1, 200, 500, 1000
- Optimal parameter:
  - 'max_leaf_nodes': 17
  - 'min_samples_split': 0.05
  - 'max_depth': None
  - 'min_sample_leaf': 1
- Accuracy = 0.827

	precision	recall	f1-score	support
0	0.81	0.85	0.83	12502
1	0.84	0.80	0.82	12570
accuracy			0.83	25072
macro avg	0.83	0.83	0.83	25072
weighted avg	0.83	0.83	0.83	25072



Random Forest – Result Analysis



Example of model failure 1 (index 7):

- Gender: female
- Age: 54
- Education level: College degree
- Country: US
- Actual performance: high
- Model prediction: low

cognitive variance!

Example of model failure 2 (index 8):

- Gender: female
- Age: 46
- Education level: Some college
- Country: US
- Actual performance: low
- Model prediction: high

Model Comparison & Conclusion

- Both models (LR and RF) successfully classify cognitive performance using demographic data
- Similar model performance: AUC = 0.9 for both LR and RF

- **Age** and **educational level** are important features for prediction in both models:
  - Higher age > low cognitive performance
  - Higher educational level > high cognitive performance
- **Gender**: little difference
- **Country**: non-predictive