Graphical user interface, application

Description automatically generatedGraphical user interface, application

Description automatically generated

Plan1.0:

1. Create new menu of iMed ✅
2. Creat the interface of log in and register(should we check the password?)
3. Finish showing objects(include dataset and models)
4. For “load dataset”, user could choose from submitted objects or upload from the computer
5. “train and store machine learning models”
6. “predict using machine learning models”
7. ….

Problem:

* On AWS, 500 internal server error (the part of prediction models works well)

Main functions:

1. **train and store models(used to be “learn prediction models”)**

there are 9 machine learning methods.

Input:

* a file with rows of samples
* target class index
* hyperparameters needed

output:

* mean\_auc\_value (used to include)

Q:

* LASSO in skit-learn has no function named predict\_proba, need to find how to plot roc\_curve in LASSO
* consider to add a button to store the model

(after finishing to train, there are two choice:

1. use this model to predict(go to 2. Use models to predict)
2. store this model(like sharing your dataset)
3. **use models to predict(used to be in “learn prediction models”)**

there are two ways:

* predict for single patient’s information (user choose list of values of features), and get predict\_label\_index
* input a file with records of patients’ information, and store all predict\_label\_index in a result file

Q: if we could store models and upload model from the user, the procedure in this part could be (1.upload or choose from existing models, 2. Choose the way to predict 3. Predict and get result)

1. **Learn epistasis**

Use **MBS** to learn interactions

Input:

* a file with rows of samples
* target class index
* alpha

Q:

* need to know what MBS is and what MBS could do

1. **learn interactive parents(used to be “learn interactions”)**

use **testweb2.jar** to learn interactive parents

Input:

* a file with rows of samples
* data column separator
* target class name
* alpha1
* alpha2
* IS threshold

Output:

* Causal sets (name is wrong)

Table

Description automatically generated

* Learned interactions

Table

Description automatically generated

* Learned nodes

Table

Description automatically generated with low confidence

1. **Test prediction**

Use testweb3.jar to do the test

Input: list all parameters, and the user select the suitable variable

Output: A picture containing logo

Description automatically generated

1. **Treatment interaction**

There are two ways: **(using testweb1.jar)**

1. System recommendation: user could input a list of values of features, the system will provide list of treatments in order to obtain the minimum probability of metastasis

Output:

Graphical user interface, text, application, email

Description automatically generated

1. User interaction: use should input not only a list of values of features, but also list of treatments, then the system will calculate the probability of metastasis

Output:

