Сравнение моделей, содержащие только положительные активности, и моделей, содержащие положительные и отрицательные активности. 5-кратная кросс-валидация. Проверка подходов к интеграции моделей.

Антон Смирнов

Apr 1, 2023

Оглавление

Описание

В предыдущем эксперименте было замечено, что наличие в модели отрицательных активностей, снижает среднюю точность модели. Для выбора окончательной модели было решено сравнить пятикратной кросс-валидацией модели, содержащие только положительные активности, и модели, содержащие как положительные, так и отрицательные активности. Подготовка данных описана в скрипте *create_datasets_epitope_mhc.R*[1]. Данные разбивались на выборки случайно в пропорции 80/20 от каждой активности с помощью пакета *caret*. Источники данных - IEDB, MHCflurry 2.0. Данные как исследования аффиности, так и масс-спектрометрии.

[1] Осторожно, при повторном запуск скриптов, обратить внимание, что нарушены правила наименования файлов!

Обработка

Общее сравнение

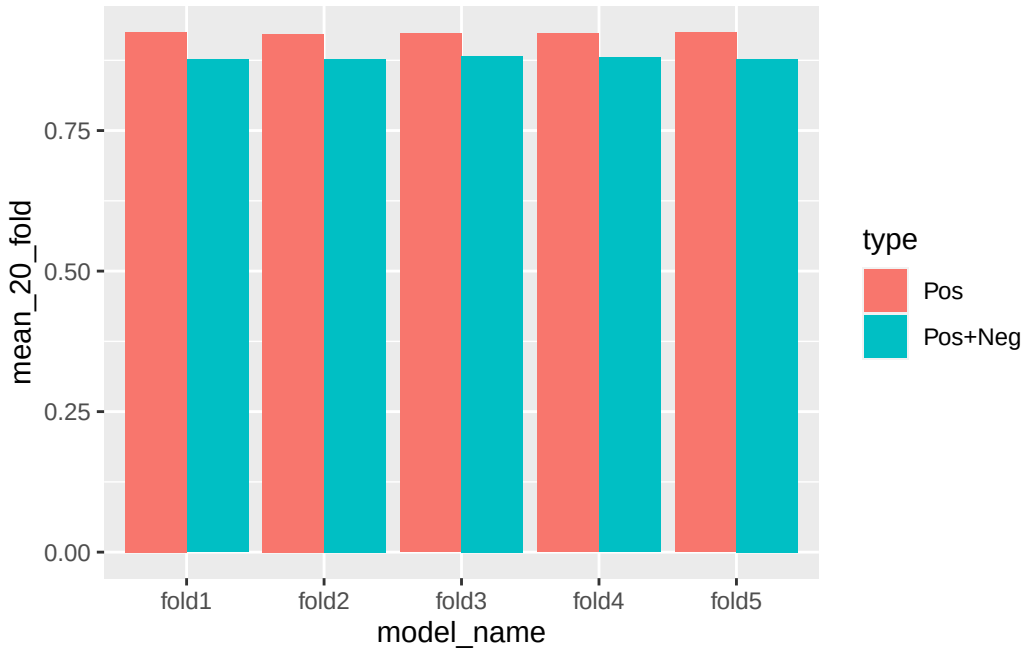
Только положительные

model_name	num_activity	mean_iap	mean_20_fold
fold1	174	0.9268655	0.9255793
fold2	174	0.9247787	0.9216862
fold3	174	0.9239299	0.9226172
fold4	176	0.9241756	0.9227295
fold5	172	0.9260081	0.9245459

И положительные, и отрицательные

model_name	num_activity	mean_iap	mean_20_fold
Fold1	278	0.8784223	0.8760446
Fold2	279	0.8797889	0.8774785
Fold3	278	0.8837644	0.8825910
Fold4	280	0.8821582	0.8804950
Fold5	279	0.8799509	0.8773333

График



Результаты 5-кратной кросс-валидации Только положительные активности

Parse results

Folds union (604162, 181)

Calculate metrics

Total activities 177

Mean AUROC 0.8781

Mean AUC-PR 0.0943

И положительные, и отрицательные активности

Parse results

Folds union (731667, 290)

Calculate metrics

!HLA-C*08:03 Only one class present in y_true. ROC AUC score is not defined in that case.

!HLA-B*27:10 Only one class present in y_true. ROC AUC score is not defined in that case.

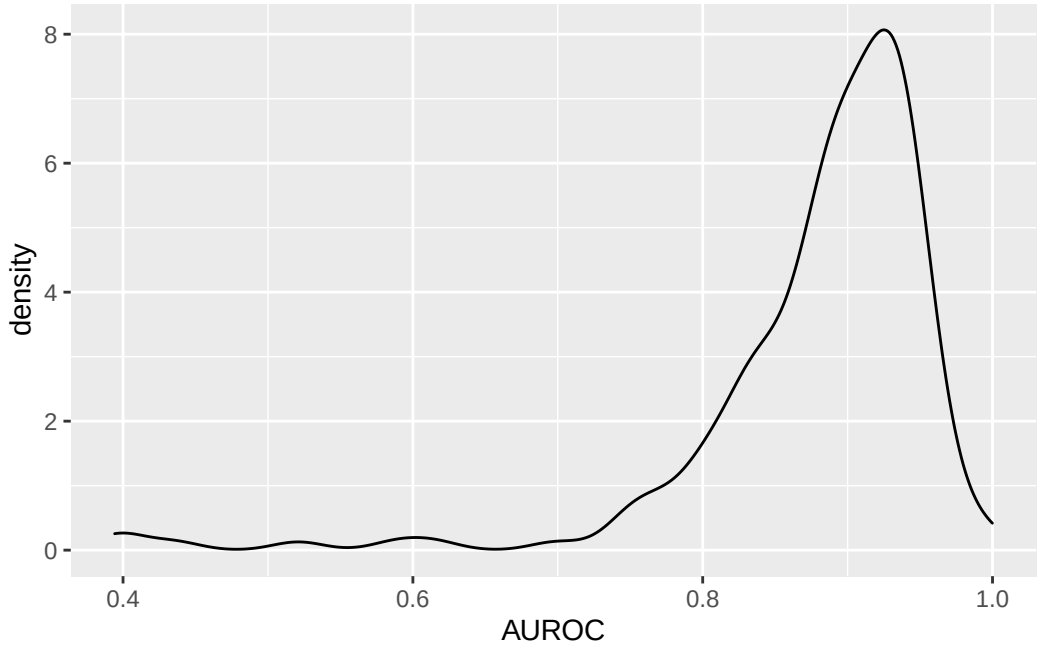
Total activities 284

Mean AUROC 0.8281

Mean AUC-PR 0.058

Только положительные

Загрузим данные



[1] "Activities with AUROC < 0.7 = 7"

[1] "HLA-B*41:05 HLA-B*39:09 HLA-B*15:18 HLA-B*51:02 HLA-B*45:06 HLA-B*35:04 HLA-B*39:10"

Средний AUC с отфильтрованными активностями

```
  mean_AUC
1   0.8928
```

10 наилучших активностей

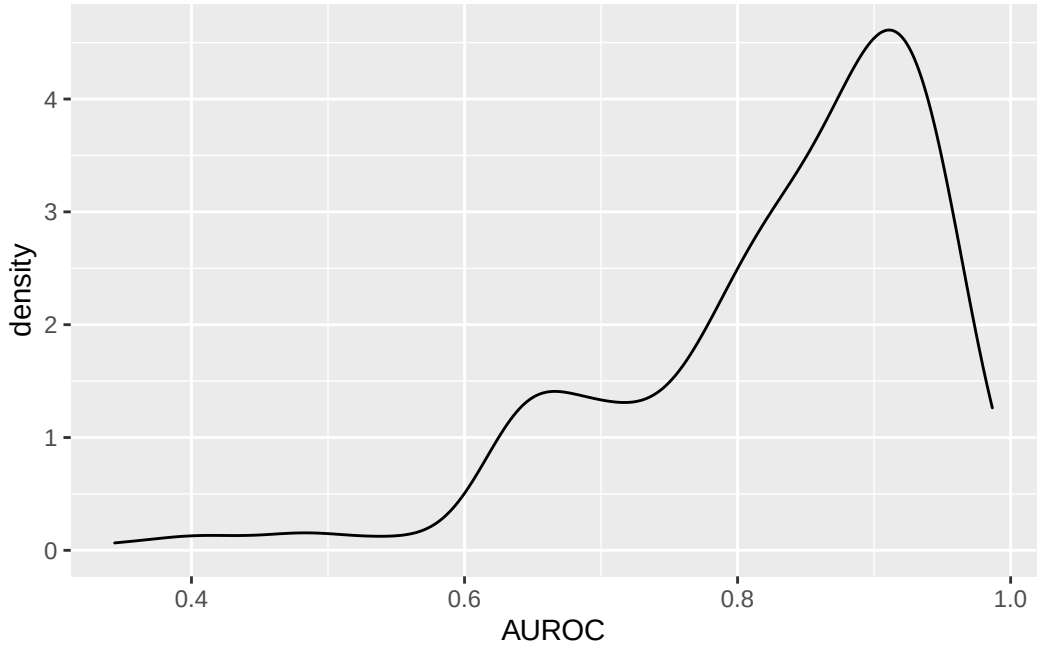
Activity	AUROC	Average.precision	num_subst	mean_iap	mean_twentyCV
HLA-B*39:05	0.9999	0.0953	7	0.9905	0.9845
HLA-A*01:03	0.9913	0.0010	6	0.9807	0.9812
HLA-B*39:06	0.9758	0.3364	802	0.9925	0.9924
HLA-A*26:08	0.9691	0.0818	329	0.9863	0.9862
HLA-B*73:01	0.9676	0.1340	236	0.9894	0.9893
HLA-B*51:08	0.9669	0.0565	500	0.9893	0.9891
HLA-B*08:02	0.9572	0.0165	30	0.9766	0.9750
HLA-B*18:03	0.9547	0.0442	184	0.9766	0.9761
HLA-B*38:02	0.9543	0.1953	2464	0.9748	0.9746
HLA-B*40:06	0.9538	0.1487	1996	0.9761	0.9759

Топ-10 наихудших активностей

Activity	AUROC	Average.precision	num_subst	mean_iap	mean_twentyCV
HLA-B*41:06	0.7937	0.0006	18	0.8720	0.8703
HLA-A*03:02	0.7929	0.0047	24	0.8279	0.8192
HLA-B*44:09	0.7894	0.0026	170	0.8268	0.8268
HLA-B*14:03	0.7733	0.0002	17	0.7994	0.7995
HLA-B*15:16	0.7691	0.0004	15	0.7910	0.7917
HLA-B*41:03	0.7681	0.0009	68	0.7827	0.7800
HLA-C*03:01	0.7596	0.0026	71	0.7800	0.7767
HLA-B*07:06	0.7523	0.0138	12	0.8731	0.8717
HLA-B*41:02	0.7484	0.0001	17	0.7706	0.7642
HLA-A*11:10	0.7424	0.0001	5	0.8637	0.8385

И положительные, и отрицательные

Загрузим данные



[1] "Activities with AUROC < 0.7 = 45"

[1] "!HLA-B*07:02 !HLA-A*03:01 !HLA-A*11:01 !HLA-B*58:01 !HLA-B*40:01 !HLA-B*15:01 !HLA-A*01:01 !HLA-B*08:01 !HLA-A

Средний AUC с отфильтрованными активностями

  mean_AUC
1   0.868

10 наилучших активностей

Activity	AUROC	Average.precision	num_subst	mean_iap	mean_twentyCV
HLA-A*01:03	0.9866	0.0006	6	0.9762	0.9727
!HLA-C*07:01	0.9783	0.0503	341	0.9859	0.9858
HLA-B*39:06	0.9739	0.3246	802	0.9926	0.9925
HLA-B*73:01	0.9689	0.1078	236	0.9893	0.9888
HLA-A*26:08	0.9669	0.0713	329	0.9862	0.9858
!HLA-C*03:04	0.9664	0.2660	161	0.9568	0.9535
HLA-B*51:08	0.9638	0.0538	500	0.9893	0.9892
HLA-B*14:01	0.9603	0.0131	14	0.9859	0.9822
HLA-B*08:02	0.9591	0.0067	30	0.9764	0.9761
HLA-B*18:03	0.9587	0.0423	184	0.9764	0.9758

Топ-10 наихудших активностей

Activity	AUROC	Average.precision	num_subst	mean_iap	mean_twentyCV
HLA-B*41:06	0.7937	0.0006	18	0.8720	0.8703
HLA-A*03:02	0.7929	0.0047	24	0.8279	0.8192
HLA-B*44:09	0.7894	0.0026	170	0.8268	0.8268
HLA-B*14:03	0.7733	0.0002	17	0.7994	0.7995
HLA-B*15:16	0.7691	0.0004	15	0.7910	0.7917
HLA-B*41:03	0.7681	0.0009	68	0.7827	0.7800
HLA-C*03:01	0.7596	0.0026	71	0.7800	0.7767
HLA-B*07:06	0.7523	0.0138	12	0.8731	0.8717
HLA-B*41:02	0.7484	0.0001	17	0.7706	0.7642
HLA-A*11:10	0.7424	0.0001	5	0.8637	0.8385

Проверка подходов интеграции оценок

```
#|warning: false
#|error: false

import pandas as pd
import os
from sklearn import metrics
from glob import glob
import numpy as np

WORKDIR = "/home/stotoshka/Documents/Epitops/PredictionEpitopes/data/cross_val/total_result_allele"
folds = glob(os.path.join(WORKDIR, "*.CSV"))
union = pd.DataFrame()
for f in folds:
    tbl = pd.read_csv(f, sep=";", header=4,decimal=",")
    union = pd.concat([union, tbl])
union = union.drop(columns=["Substructure Descriptors","New Descriptors","Possible Activities at Pa > Pi"])
union = union.rename(columns = {"<activity>":"activity"})
activities = union.columns[1:]
prediction = union.query("activity in @activities")
negative_activities = sorted([a for a in activities  if "!" in a])
positive_activities = sorted([a for a in activities  if "!" not in a and "!"+a in negative_activities])
total_train_data = pd.read_excel("/home/stotoshka/Documents/Epitops/PredictionEpitopes/data/cross_val/total_result_allele
```

<string>:1: FutureWarning: Indexing with multiple keys (implicitly converted to a tuple of keys) will be deprecated, use a list inst

```
result = pd.DataFrame(columns=["Activity","AUROC", "Average precision"])
for i, (pos, neg) in enumerate(zip(positive_activities,negative_activities)):
    pred = np.where((prediction.loc[prediction[pos].notnull() & prediction[neg].notnull(),pos] > 0) | (prediction.loc[prediction[pc
```

```
    true = np.where(prediction.loc[prediction[pos].notnull() & prediction[neg].notnull(),"activity"] == pos, 1, 0)
    roc_auc = metrics.roc_auc_score(true, pred)
    pr_auc = metrics.average_precision_score(true, pred)
    result.loc[i] = [pos + "/" +neg, roc_auc, pr_auc]

result1 = pd.DataFrame(columns=["Activity","AUROC", "Average precision"])
for i, (pos, neg) in enumerate(zip(positive_activities,negative_activities)):
    pred = np.where((prediction.loc[prediction[pos].notnull() & prediction[neg].notnull(),pos] > 0) | (prediction.loc[prediction[pc
    true = np.where(prediction.loc[prediction[pos].notnull() & prediction[neg].notnull(),"activity"] == pos, 1, 0)
    roc_auc = metrics.roc_auc_score(true, pred)
    pr_auc = metrics.average_precision_score(true, pred)
    result1.loc[i] = [pos + "/" +neg, roc_auc, pr_auc]
```

Area under ROC

Усреднение 0.8406

Взвешенное по 20CV 0.8529

Area under Precision-Recall curve

Усреднение 0.057

Взвешенное по 20CV 0.0631

Выводы

1. Модели с использованием только положительных активностей дают большую точность по результатам 5-кратной, 20-кратной и leave-one-out кросс-валидации.
2. Модели имеют хороший AUROC, но крайне низкий AUC-PR.
3. Взвешивание по 20-кратной кросс-валидации дает большую точность прогноза, чем простое усреднение.