

2019年度 卒業論文

rwhoを用いた離学者トリアージシステム

大阪産業大学 デザイン工学部 情報システム学科
情報教育システム研究室

16H010 石本七星

rwwho を用いた離学者トリージシステム

16H010 石本七星

1 はじめに

近年、大阪産業大学(以下、本学という)では離学者数が問題とされている。デザイン工学部情報システム学科だけでも年間 10% 程度の学生が離学者になっている。そこで本学科のプログラミング演習室(以下、演習室という)で記録されている rwwho^{*1}の情報を基に、必修科目の出席情報から離学者となる学生の傾向の割り出しを試みた。本研究では、過去 4 年分のログイン情報を基に離学者になる可能性のある学生の傾向の割り出しを行い、その傾向を基に離学者になる可能性のある学生を危険度別にメールで通知するシステムの開発を行った。

2 目的

本研究の目的は、離学者になる可能性のある学生を早期の段階で発見し、教員に通知するシステムを開発することである。

3 離学者防止システムの概要

本研究では、演習室でのログイン情報と講義情報から各学生の出席状態を把握し、その傾向に基づいて、離学者になる可能性がある学生を危険度別に分ける。各学生のログイン情報は、演習室で rwwho を用いて記録しているため、これを用いる。講義情報は、講義の予定を Google Calendar で管理し、API を用いて情報を取得する。危険度別に分ける方法としては、離学者になるであろう学生の傾向を基に、医療現場で使われるトリージカテゴリー [1] を使用し離学トリージカテゴリーとして振りわけた。本来のトリージカテゴリーと同様に色は RED、YELLOW、GREEN、BLACK に危険度を分け、メールを教員に通知する。本研究では GREEN の学生が急に RED や BLACK に危険

度が進むことはごく少数であり、殆どは GREEN、YELLOW、RED、BLACK と順に進む。GREEN から YELLOW へ上がった時点でメールにて教員に通知が行われる。その他、カテゴリーに変更がある場合に通知が行われる。

4 検証結果

検証には、15H から 18H までの学生の情報を基に作成した 3 つの数値化モデルを用いた。19H の前期の出席情報をこのモデルを用いて検証を行った。数値化モデルには加点する場所を変えたモデル A、モデル B、モデル C を作成、妥当性を確かめるために比較を行った。その結果、本研究の目的となる離学者になる可能性のある学生の早期発見の為、判定までの精度が高かったモデル C を使用することで、15 回の授業を全て休んでいる学生なら 2 回目で RED 判定、4 回目で BLACK の判定が出る結果となった。モデル C は連続した欠席と、授業の 1 回目と 2 回めの欠席を重視したモデルである。

5 まとめ

本研究では、本学の離学者の数が問題となっていることから、離学者になるであろう学生を自動判断し、メールにて教員に通知する離学者トリージシステムを開発した。

19H の学生の前期のデータで検証するために作った 3 つのモデルのうち、判定の精度が高かったモデル C を使用することで、学生が色別・危険度別に振り分けられる。その結果、このシステムを使用することで、最優先に対策しなければいけない学生が明確にわかり、教員の支援と早めの対策ができると考えている。

参考文献

- [1] 篠原拓哉. 災害時のトリージの現状-救急医療の現状と課題(後編). 2016.

^{*1} ローカルネットワーク上の全てのマシンにログオンしたユーザーのユーザー名、ホスト名、各セッションの開始日時を表示する。Posix の command。

目次

1	はじめに	1
2	目的	2
3	分析対象	3
4	分析方法	3
4.1	ログイン情報	4
4.2	成績情報	4
4.3	講義情報	5
5	傾向の考察	6
5.1	連続で休んでいる学生	7
5.2	3回以上休んでいる学生	8
5.3	授業欠席のターニングポイント	10
5.4	1回目と2回目の授業を連続して休んでいる学生	11
5.5	不良者と離学者の関係	12
6	離学トリアージカテゴリー	16
7	離学者トリアージシステム	17
7.1	システムの概要	17
7.2	システム詳細	17
7.3	新規登録機能	19
7.4	出席確認機能	19
7.5	数値化機能	21
7.6	リストアップ機能	22
7.7	メール送信機能	23
8	検証	24
8.1	傾向の数値化	24
8.1.1	モデル A	25
8.1.2	モデル B	26
8.1.3	モデル C	27
9	結果と考察	28
9.1	各モデルの妥当性の考察	28
9.1.1	最初から一度も出席していない学生	29
9.1.2	途中から一度も出席しなくなった学生	29
9.1.3	所々欠席があり、最終的に出席しなくなった学生	30
9.1.4	結論	30
10	まとめ	31

10.1	今後の課題	31
付録 A	ソースコード	34
A.1	数値化プログラム	34
A.2	メール送信プログラム	36

1 はじめに

近年、大阪産業大学(以下、本学という)では離学者数が問題とされている。デザイン工学部情報システム学科(以下、本学科という)だけでも年間10%程度の学生が離学者になっている。

そこで、rwho^{*1}の情報と過去のデータより割り出した傾向を基に、離学者になる可能性のある学生を自動判別して通知するシステムを開発することで、離学者の早期発見を目指した。

本システムは本学科のプログラミング演習室(以下、演習室という)で記録されているrwhoの情報を基に、必修科目の出席情報から離学者となる学生の傾向の割り出しを試みた。本研究でrwhoの情報として使用しているのは、ログイン名とログイン・ログアウトの時刻情報である。また、過去4年分の離学者の成績情報と退学時期から、離学者の約95%は成績不良者になってから退学しており、その中でも44%の学生が成績不良者になってから半年後に退学していることがわかった。つまり、離学者の退学理由には成績不良が大きく関係しており、早期の段階で当該学生にアプローチをかけることが重要になると考えられる。

本研究では、過去4年分のログイン情報を基に離学者になる可能性のある学生の傾向の割り出しを行い、その傾向を基に離学者になる可能性のある学生を早期に発見し、危険度別にメールで通知するシステムの開発を目的とする。

第2章では本研究の目的について述べる。第3章では本研究で行った傾向の分析対象について述べる。第4章では傾向の分析方法について述べる。第5章では本研究で発見した傾向について述べる。第6章では傾向を基に作成した離学トリアージカテゴリーについて述べる。第7章では本研究で開発したシステムの概要を述べる。第8章では本研究で行った検証の結果を述べる。第9章では検証の結果をその考察とともに述べる。第10章には研究の成果とともに今後の課題についてまとめる。

^{*1} ローカルネットワーク上の全てのマシンにログオンしたユーザーのユーザー名、ホスト名、各セッションの開始日時を表示する。Posixのcommand。

2 目的

本研究の目的は、過去4年分のログイン情報を基に離学者になる可能性のある学生の傾向の割り出しを行い、離学者になる可能性のある学生を早期の段階で発見し、危険度別に学生をメールにて教員へ通知するシステムを開発することである。

現在は成績不良の学生に対して半年に一度の呼び出しが行われている。しかしそれでは、成績不良者は半年後にしか分からず、単に修得単位数の少ないという情報しかないため危険度がわからないのが問題となっている。

そこで本システムを使用することにより、最優先に対策しなければいけない学生が早期に発見でき、問題解決の一步になると考えている。

3 分析対象

本研究では学生を rwho から得られる情報と学生を離学者・不良者・優良者に分けたデータを比較し、どういった学生が離学の傾向にあるのかを調べた。対象は以下の 2 つである。

- 15H から 18H の学生
- 1 年次と 2 年次の演習室で行われる必修科目

1 年次と 2 年次に演習室で行われる必修科目は、1 年次の前期は月曜 5 限のデジタルコンテンツ演習。後期は月曜 3 限のプログラミング 2。2 年次の前期は木曜 4・5 限のプログラミング演習 1。2 年次の後期は rwho のシステムに問題があった可能性があり、rwho の情報がうまく得られていなかったため、その期間のデータ使用不可と判断した。また、18H の学生は 1 年次の前期後期のみのデータから傾向を発見した。

各学生の説明は以下の通り定義する。

- 離学者
現時点のデータで本学を離学していった学生のことを表す。
- 不良者
年間で修得単位が 33 単位未満のものを表す。
- 優良者
年間で修得単位が 33 単位以上のものを表す。

本研究では学籍番号は個人を特定できないよう hash 化されている。本論文では「s16H67d34c65」のように表示されている。

4 分析方法

本研究では、傾向の分析にログイン情報、成績情報、講義情報から作成した出席情報を使用する。rwho から取得したログイン情報と講義が行われた時間とを比較して出席情報を出すプログラムを通して CSV ファイルに出力される。各回、出席なら 2、遅刻なら 3、欠席なら 0 と表記される。実際にプログラムを実行し出力された CSV ファイルの例を図 1 に示す。

```
s16H51b31e5e,2,2,2,2,2,2,2,2,2,2,0,0,2,2,2,2,2
s16H53df8f3a,2,2,2,2,2,2,2,2,2,2,2,2,0,2,2,2,2
s16H5e83361c,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,0
s16H5fa2df15,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2
s16H67cb025b,2,2,2,2,2,2,2,2,2,2,2,0,2,2,2,2,2,2
s16H67d34c65,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2
s16H714da10b,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2
s16H73380395,2,2,2,2,2,2,2,2,2,2,2,2,3,2,2,2,2,2
s16H7dacc974,2,2,2,2,2,0,0,0,0,0,0,0,0,0,0,0,0,0
s16H823898d1,2,2,2,2,2,2,2,0,2,0,0,0,0,0,0,0,0,2
```

図 1 対象の学生の出席情報が記録された CSV ファイルの例。最初のコンマまでは学籍番号を、その後のコンマは 1 回目から 15 回目までの授業の出欠情報を示している。各回、出席なら 2、遅刻なら 3、欠席なら 0 が表記されている。

4.1 ログイン情報

演習室では端末のログイン情報を、`rwho` を用いたシステムにより取得、記録している。これによって、誰が、どの端末で、どのターミナルで、いつから、いつまで利用していたということがわかる。ログファイルに出力されたログイン情報の例を図 2 に示す。

```
s19H09854fba@WS02:win1 2019/04/09 10:48 - 2019/04/09 10:51
s19H584d9aaa@WS22:win1 2019/04/09 10:39 - 2019/04/09 11:13
s19H36ad1f63@WS32:win1 2019/04/09 10:39 - 2019/04/09 11:19
s19H6c772c80@ES17:win1 2019/04/09 10:43 - 2019/04/09 11:33
s18H4eb041cc@ES15:win1 2019/04/09 10:51 - 2019/04/09 11:44
s18Hd36fb6ac@WS74:win1 2019/04/09 10:33 - 2019/04/09 11:59
s19He22137a2@WS13:win1 2019/04/09 11:28 - 2019/04/09 12:00
s18Hee636efa@ES11:win1 2019/04/09 10:41 - 2019/04/09 12:01
s19H51a950fa@WS12:win1 2019/04/09 11:26 - 2019/04/09 12:01
s19Hb6f49ad0@WS33:win1 2019/04/09 11:26 - 2019/04/09 12:01
```

図 2 ログイン情報が記録されたログファイルの例。学籍番号@端末番号:ターミナル名^{*2}ログインした日付と時刻-ログアウトした日付と時刻の順に表されている。これにより、誰がどの端末で、いつからいつまで利用していたかが分かる。

4.2 成績情報

本研究を行うにあたり、成績不良者と離学者のリストを使用した。リストの内容は、誰が何年の前期・後期どちらに成績不良になったか、退学したかが記載されている。実際に使用した成績情報の例を図 3 に示す。

```
s16H67d34c65, 2018A,
s16H73380395, 2016A,
s16H84a2988b, 2016B,
s16Hf395b5de, 2016A, 2017A
s16H2a41b05c, 2016B,
s16Hc70a976f, 2016A, 2017A
s16H5fa2df15, 2016A,
s16Hec544f86, 2016B, 2018A
s16Hbdafe1d7, 2016B,
s16He72a8ee3, 2016A, 2017A
```

図 3 成績不良になった時期と、退学した時期が記載されたテキストファイルの例。この図は学籍番号、成績不良になった年、離学した年の順に表されており、年度の後ろについた A は前期、B は後期を示している。

^{*2} Win1 は Windows を示している。

4.3 講義情報

過去の rwho のログイン情報から出席情報を取得するには、過去にいつ講義が行われたのかの情報が必要となる。本研究では、本大学の学年暦からいつ講義が行われたのかを取得している。しかし、学年暦からのみの情報では実際に休講であったとしても講義が行われたと処理され、殆どの学生は欠席として判定される。そのため、欠席率が不自然に高い回であった場合、その回は例外と処理し休講と判断する。欠席率が不自然に高くなった回が有る講義を例として図 4 に示す。

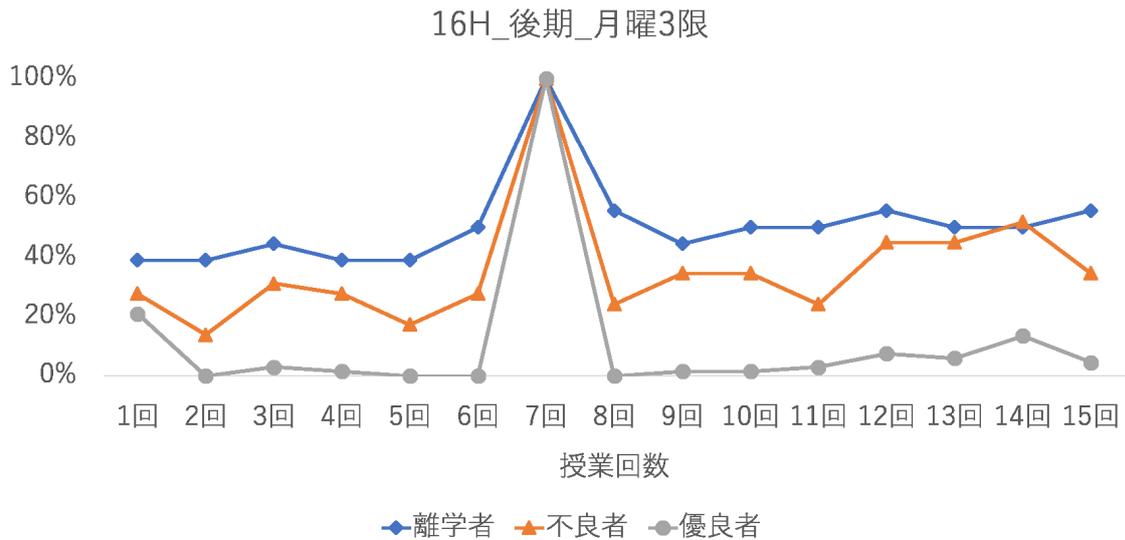


図 4 学年暦では講義があったが、欠席率が不自然に高くなった回がある講義の例。この図は 16H の後期、月曜 3 限に行われるプログラミング 2 の欠席の割合を示したグラフである。縦軸は欠席割合を、横軸は授業の回数をそれぞれ示す。また、 離学者 は不良者、 優良者をそれぞれ示す。このグラフから 7 回目の授業がどの学生も欠席率が 100% を示している。結果、7 回目の授業は例外処理として、休講だったと判断する。

5 傾向の考察

本研究では、過去4年分の離学者の成績情報と rwho のデータを基に、離学者になるであろう学生の傾向を調べた。発見した傾向は以下の5つである。

- 連続して休んでいる学生
- 3回以上休んでいる学生
- 授業欠席のターニングポイント
- 1回目と2回目の授業を連続して休んでいる学生
- 不良者と離学者の関係

以下に傾向の詳細を記載する。

5.1 連続で休んでいる学生

離学者は連続で授業を休んでいる。全年度*3を足した学生の合計数は1226名いる。その中で2連続休んでいる学生が255名、3連続が168名、4連続が141名、5連続が126名いる。その学生らを連続別、各学生別*4に割合を出した。割合は式(1)で求める。

$$x = \text{連続回数別の各学生の合計} / \text{連続回数別の学生合計} \quad (1)$$

上記の式で出した割合から、優良者で2回連続休んでいる学生は19%。また、3回連続休んでいる学生は7%しかない。これに対し、離学者で2回連続休んでいる学生は44%。3回連続休んでいる学生は61%いることがわかっていて、また、離学者は3連続以降でも約60%の学生が連続して休んでいる。その結果を図5に示す。

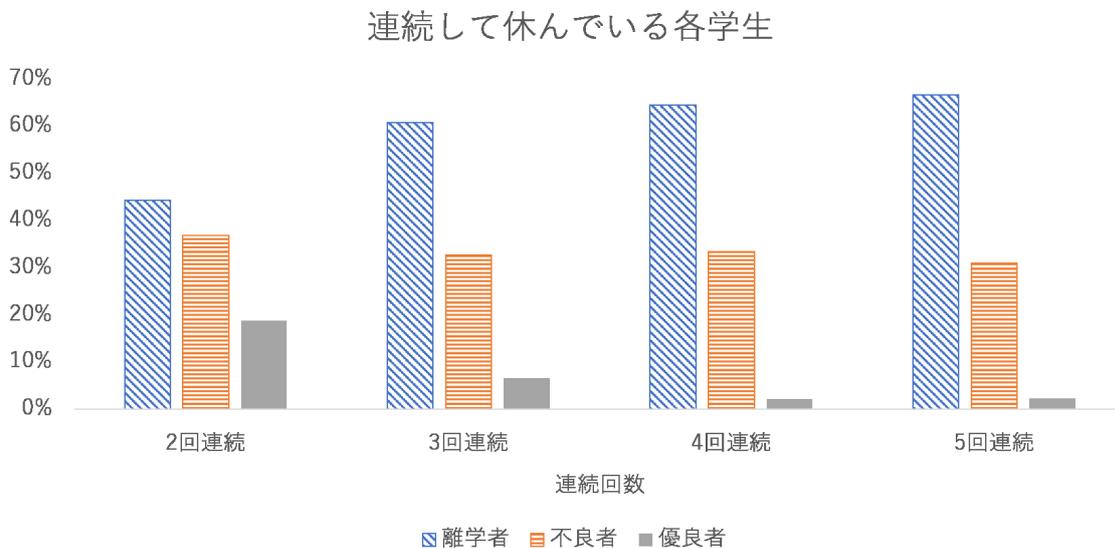


図5 2連続から5連続までの連続して休んでいる学生を表示。斜線は離学者を、横線は不良者を、塗りつぶしは優良者をそれぞれ示している。割合は式(1)で求める。この割合から、優良者は2連続目では19%いるが、3回連続目以降に連続して休んでいる学生は10%も居ない。それに対し、離学者は2連続目では44%、3回目以降は約60%以上者学生が連続して休んでいることが分かる。

*3 全年度とは、18Hの1年次前期後期と15Hから17Hの1年次の前期後期、2年次の前期を指す。

*4 各学生とは離学者・不良者・優良者の事を示す

5.2 3回以上休んでいる学生

15回の授業の中、各学生が何回休んだか。ここでの割合は全年度の休んだ回数ごとの学生合計数から出しており、式(2)で求める。

$$X = 15 \text{ 回中何回休んだか (休んだ回数ごと) / 休んだ回数ごとの学生合計数} \quad (2)$$

また、休んだ回数ごとの人数は下記の表6に示す。

	単位:人			
	離学者	不良者	優良者	合計
1回休んだ	143	191	266	600
2回休んだ	124	147	123	394
3回休んだ	113	107	51	271
4回休んだ	109	81	21	211
5回休んだ	99	62	11	172
6回休んだ	95	49	6	150
7回休んだ	91	45	6	142
8回休んだ	86	41	3	130
9回休んだ	83	38	2	123
10回休んだ	79	34	1	114
11回休んだ	74	27	1	102
12回休んだ	68	26	1	95
13回休んだ	57	20	0	77
14回休んだ	39	13	0	52
15回休んだ	23	6	0	29

図6 休んだ回数ごとの各学生の合計人数と休んだ回数ごとの合計を表示。

その割合結果から、優良者では1回や2回休んでいる者は30%以上いるが、3回目から20%を下回り19%の学生しかいない。これは767人中約15人に1人の計算である。しかし、3回目の時点で離学者は42%、不良者は39%の学生がいる。これは離学者172人中約2人に1人、不良者は287人中約3人に1人いる。このことから、3回授業を休む学生を数値化を行う際の加算ポイントとする。その結果を図7に示す。

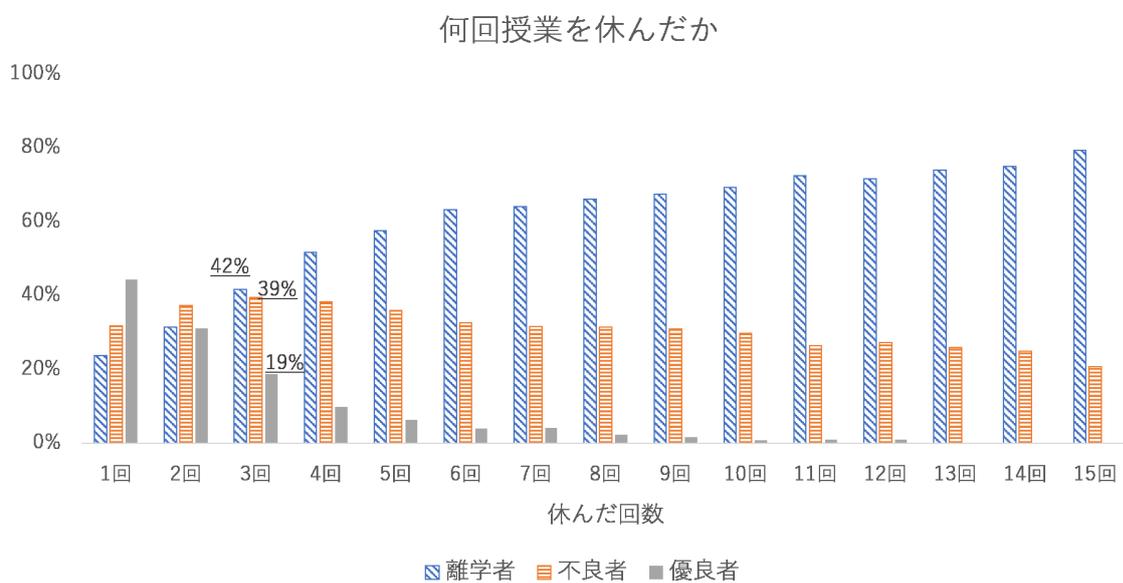


図 7 15 回の授業中で各学生が何回休んでいるかの割合を見たもの。割合は式 (2) で求める。斜線は離学者を、横線は不良者を、塗りつぶしは優良者をそれぞれ示している。優良者では 1 回や 2 回休んでいる者は 30% 以上いるが、3 回目から 20% を下回り 19% の学生しかいない。これは 767 人中 15 人に 1 人の計算である。しかし、3 回目の時点で離学者は 42%、不良者は 39% の学生がいる。これは離学者 172 人中 2 人に 1 人、不良者 287 人中 3 人に 1 人いる。その結果から、3 回授業を休む学生を数値化を行う際の加算ポイントとして考える。

5.3 授業欠席のターニングポイント

1 回から 15 回目までの授業の、離学者は来なくなっているターニングポイントが無いか、そのターニングポイントは何回目かを調べた結果。殆どの離学者が 3 回目から上がってきていることがわかった。割合は式 (3) で求める。

$$x = 15 \text{ 回授業が有るうちの } n \text{ 回目に欠席にした全年度の各学生数/全年度の各学生の合計} \quad (3)$$

全年度の各学生の合計は、離学者 172 名、不良者 287 名、優良者 767 名のことを示す。その結果を図 8 に示す。

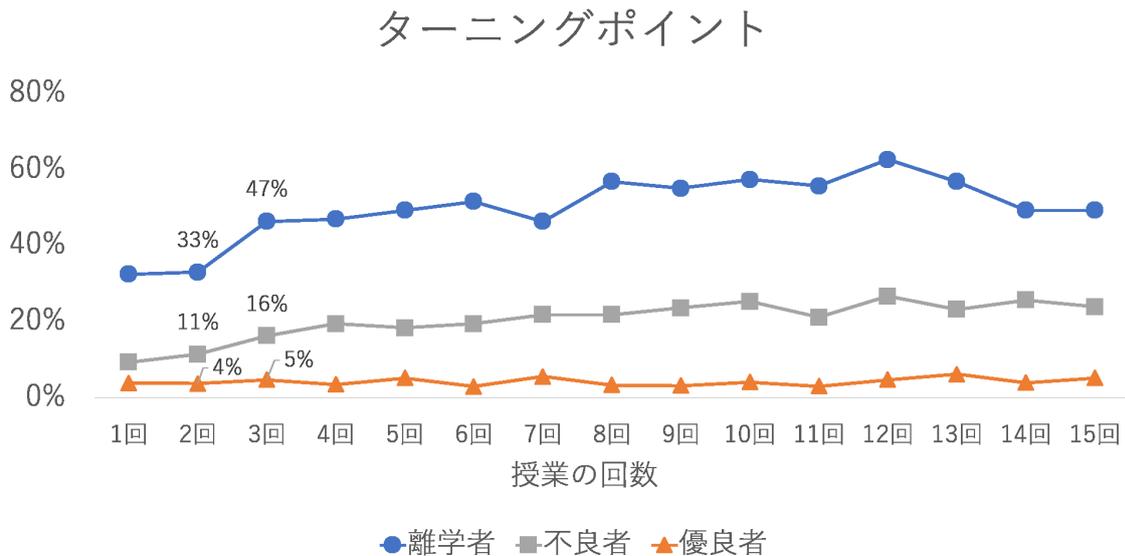


図 8 離学者が欠席し始めるターニングポイント。 が離学者、 が不良者、 が優良者をそれぞれ示している。割合は式 (3) で求める。それを比較し、何回目の授業から離学者は休みだしているかターニングポイント調べた。結果、3 回目の授業で離学者は前回の授業より 14% ポイント上がった 47% の学生が休みだしており、ここにターニングポイントがあると考えた。

5.4 1回目と2回目の授業を連続して休んでいる学生

全体で1回目と2回目の授業を連続して休んでいる学生は172人中61人おり、離学者は47名、不良者は13名、優良者は1名いる。そのうち77%の割合を離学者が占めており、その次に21%が不良者、残り2%が優良者となっている。割合は式(4)で求める。

$$X = \text{各学生}/\text{合計人数 (61名)} \quad (4)$$

その結果を図9に示す。

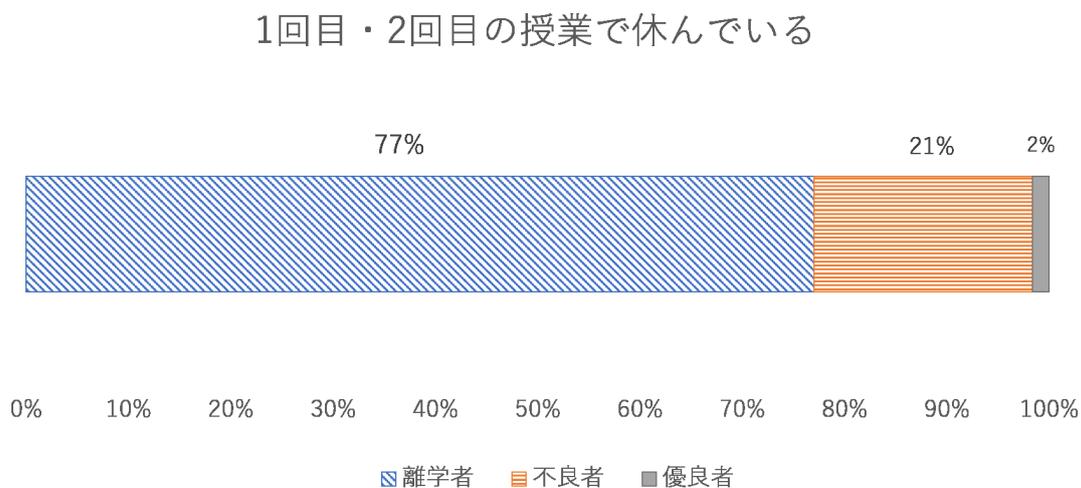


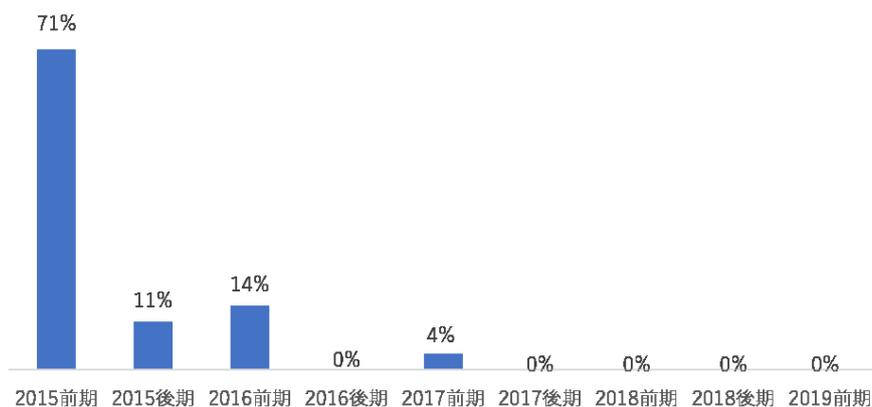
図9 1回目と2回目の授業を連続して休んだ学生。15Hから18Hの1年次、前期後期と15Hから17Hまでの2年次前期で、1回目と2回目の授業を連続して休んでいる学生の合計は61名。その中で離学者は47名、不良者は13名、優良者は1名。斜線は離学者を、横線は不良者を、塗りつぶしは優良者をそれぞれ示す。割合は式(4)で求める。

5.5 不良者と離学者の関係

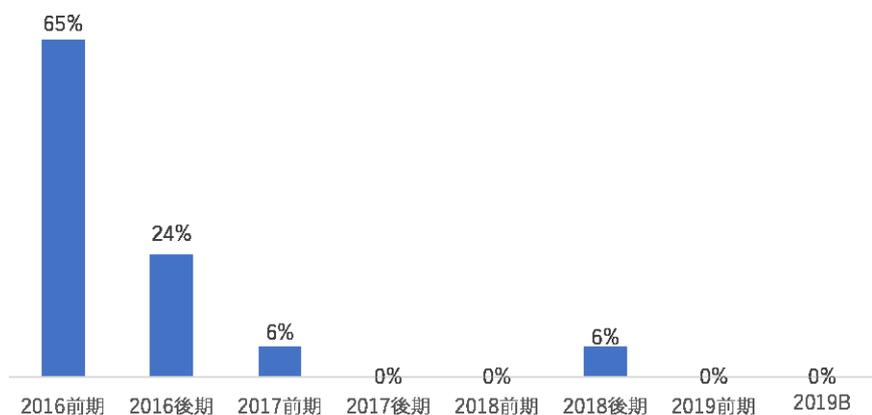
入学時の学生の合計は、15H が 117 名、16H が 114 名、17H が 114 名、18H が 108 名のうち、離学者は 68 名。その 68 名中、一度不良者になってから離学している学生は 65 名いる。その殆どの学生が不良になった時期が 1 学年次の前期に 60% 以上の学生が不良者となっており。その一年後に 30% 以上の学生が辞めていることが本研究でわかっている。離学者が成績不良になるまでの図を図 10 に、離学者が辞めた年の図を図 11 にそれぞれ示す。

この結果から離学者には本人の決意はともかく、引き止めることができる期間が半年あることがわかった。そのため、何らかのアプローチを強化することにより離学者の数は減らすことができるのでは無いかと考えた。

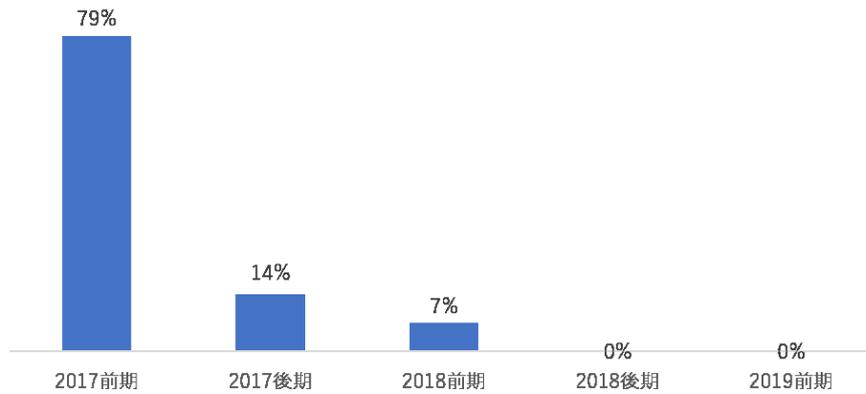
15H 離学者が不良になった年



16H 離学者が不良になった年



17H 離学者が不良になった年



18H 離学者が不良になった年

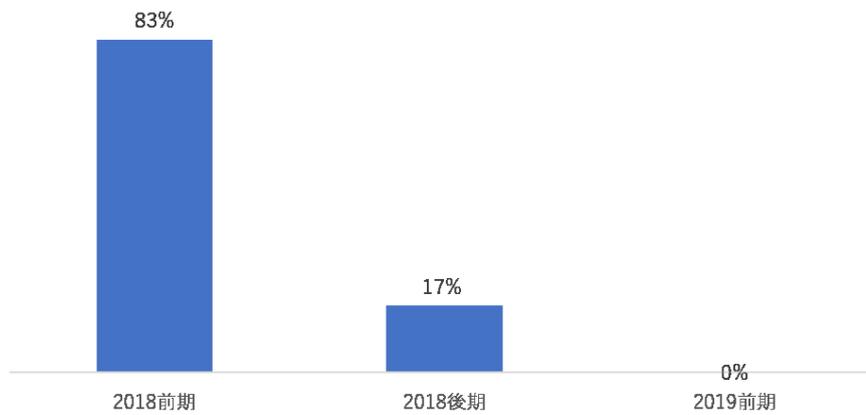
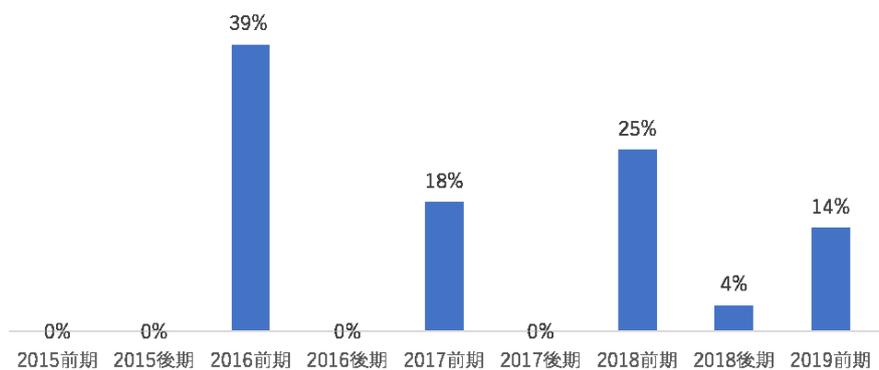
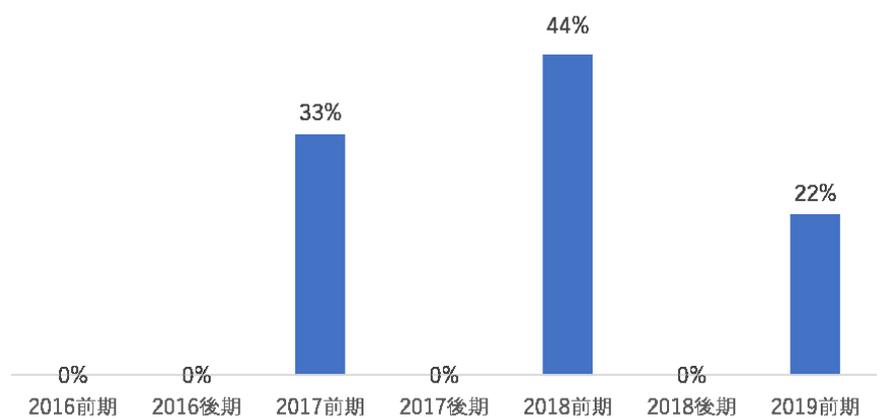


図 10 15H から 18H までの離学者が不良者になった年。この図は縦軸が不良になった離学者の割合、横軸が年度を表しており、棒グラフ自体は離学者が不良者になった年を表している。15H から 18H までの離学者の合計は 68 名。そのなかで 65 名の学生が一度不良者になってから離学している。また、その 65 名の中でも、60% 以上の学生が 1 年次の前期に不良者になっている。

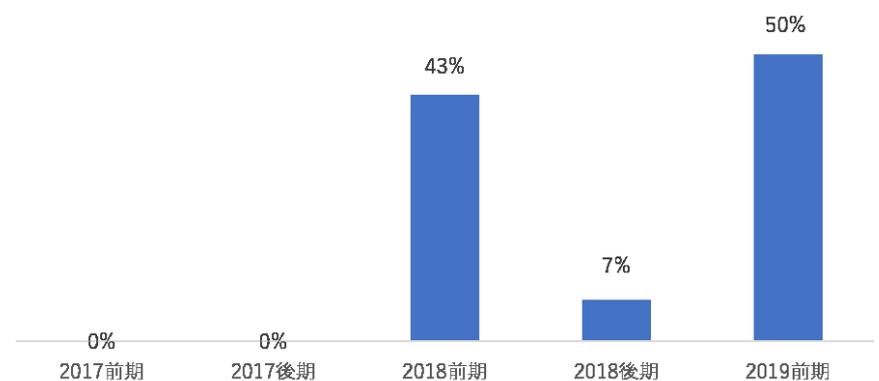
15H 離学者が辞めた年



16H 離学者辞めた年



17H 離学者が辞めた年



18H 離学者が辞めた年

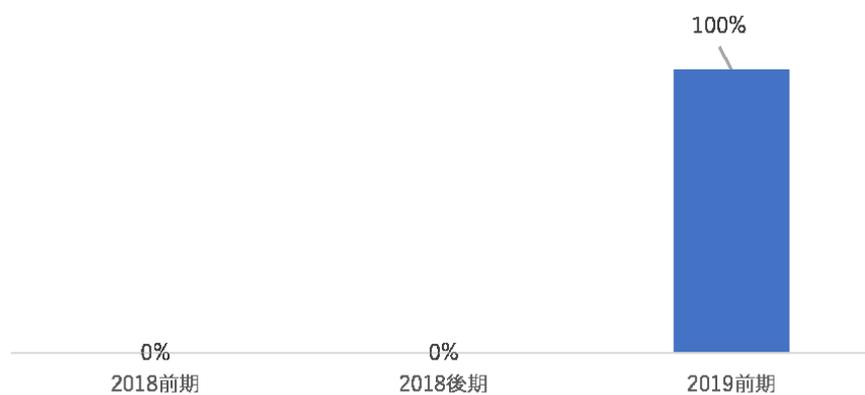


図 11 15H から 18H までの離学者が辞めた年。この図は縦軸が辞めた離学者の割合、横軸が年度を表しており、棒グラフ自体は離学者が辞めた年を表している。図 18 で示しているとおり離学する生徒の大半が 1 回生の前期に離学している。それに対し、離学していく学生は 2 年の前期か 3 年の前期に偏っていることが分かる。これにより、早期発見することで、離学者への指導を早められ、対策に繋ぐことができると考えている。

6 離学トリアージカテゴリー

本研究では、医療現場で使われるトリアージカテゴリー [1] を使用し、離学トリアージカテゴリーとして学生を振り分けた。本来のトリアージカテゴリーと同じで色は RED、YELLOW、GREEN、BLACK に分かれて判別している。RED は何らかの対策を直ちにしなければ、離学・成績不良により卒業できないもの、最優先的に対策していかなければいけない生徒のことを示す。YELLOW は多少対策が遅れても着実に卒業できるであろうもの、RED よりは対策を早めなくても良い事を示す。GREEN はほとんど手を付けなくても卒業して行くであろうもの、対策をしなくても安全な生徒のことを示す。BLACK は手を付けても仕方がないもの、どうしても離学・成績不良は免れないであろうものを示す。

本研究では GREEN の学生が急に RED や BLACK に危険度が進むことはごく少数であり、殆どは GREEN → YELLOW → RED → BLACK と順に進む。また、GREEN から YELLOW へ上がった時点でメールにて教員に通知が行われる。その他、カテゴリーに変更がある場合に通知が行われる。

下記に例として離学トリアージカテゴリーを図 12 に示す。

	分類	識別色	状態	具体的な事例
第一順位	最優先対策群	赤色	何らかの対策を直ちにしなければ、離学・成績不良により卒業できないもの	4回休んでいる。3回連続で休んでいる。1回目・2回目を連続して休んでいる。
第二順位	対策群	黄色	多少対策が遅れても、着実に卒業できるであろうもの	3回休んでいる。2回連続休んでいる。3回目の講義を休んでいる。
第三順位	安全群	緑色	ほとんど手を付けなくても卒業して行くであろうもの	初期段階
第四順位	死亡群	黒色	手を付けても仕方がないもの、どうしても離学・成績不良は免れないであろうもの	5回休んでいる。4回以上連続で休んでいる。

図 12 離学トリアージカテゴリー、上記の表にそって離学者になるであろう学生の危険度を色に表して識別している。

7 離学者トリアージシステム

7.1 システムの概要

本研究では、演習室でのログイン情報と講義情報から各学生の出席状態を把握し、その分析した傾向に基づいて、離学者になる可能性がある学生を危険度別に分ける。各学生のログイン情報は、演習室で rwho を用いて記録しているため、これを用いる。講義情報は、講義の予定を Google Calendar で管理し、API*5 を用いて情報を取得する。本システムの大まかな流れを図 13 に示す。

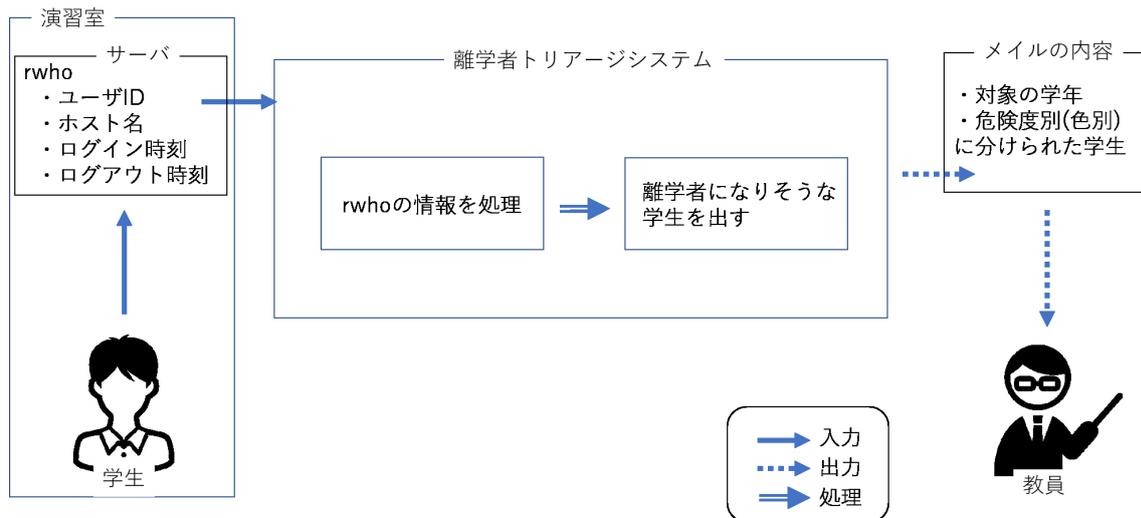


図 13 本システムの大まかな処理の流れである。演習室のサーバに学生がログインすることで、rwho から情報が取れ、その情報を処理し離学者通知システムが動き、教員にメールという形で危険度別に別れた学生の情報が届く。

7.2 システム詳細

章 7 では図 13 に書かれている離学者トリアージシステム内の詳細を記載する。本システムは Shell*6 Script*7 を使用し大きく 5 つに分けて機能を作成した。

1. 新規登録機能

各期間中 (年度の前期後期) にログイン情報に対象の学生 (15H から 18H まで学生) が初めて出現した場合に、データベースに学生の情報を新規登録する。

2. 出席確認機能

Google Calendar の講義予定の情報と、ログイン情報を比較し出席情報を出力する。

3. 数値化機能

各学生の出席情報を読み取り、それぞれに点数をつける。

4. リストアップ機能

各学生の点数と、危険度別のしきい値を比較しリストに出力する。

*5 ソフトウェアなどの一部を外に向けて公開し、第三者が開発した機能と共有できるようにしてくれるインターフェースの一つ

*6 コンピュータの OS を構成するソフトウェアの一つ。利用者からの操作の受付や、利用者への情報の提示などを担当する。

*7 Shell Script とは UNIX 系 OS を操作するためのシェル上で実行できる簡易のプログラミング言語。

5. メール送信機能

リストアップ機能で出力したリストを確認し、教員宛にリストを送信する。

離学者通知システムの流れを図 14 に示す。

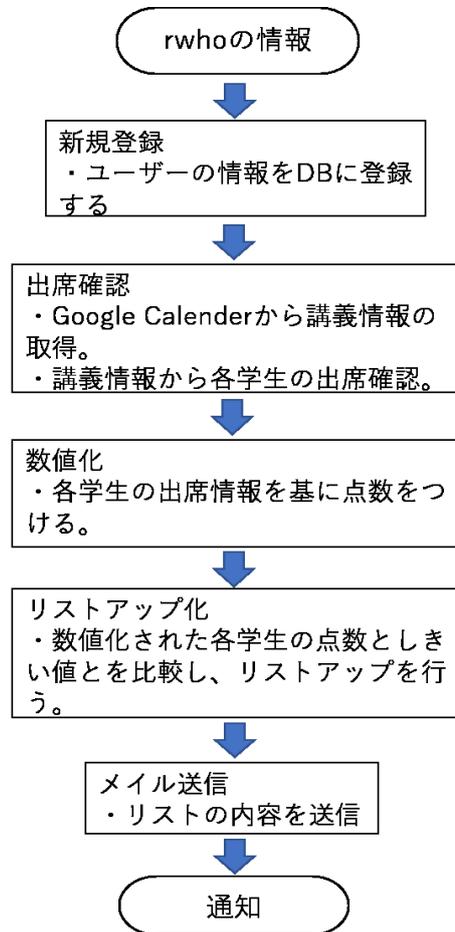


図 14 離学者通知システムの詳細を示した図。

7.3 新規登録機能

学生の点数や危険度はデータベースで管理されている。その学生の情報は、年度と半期毎で作成する。そのため、年度と半期中に初めてログイン情報に記録された学生がいた場合に、新規にデータベースへ、その学生の情報が作成される。

7.4 出席確認機能

出席の確認は rwho から取得したログイン情報と、講義が行われた時間とを比較することで行う。しかし、講義の情報を取得するにあたり、休講や補講などによる予定の変更が生じる可能性が有る。そこで、教員が簡単に講義の予定に変更を行えるよう、本研究では Google Calendar を使用して予定を管理している。Google Calendar のイベントの取得には Google Calendar API を使用する。

Google Calendar API では、ユーザが条件を指定してイベントを取得することができる。Google Calendar API の使用には、登録時に発行される API_KEY と、参照したいカレンダーの CALENDAR_ID を指定。取得したいイベントの条件を Curl^{*8} で Google Calendar API の URL に GET することで JSON 形式^{*9}で結果が返ってくる。本システムでは、指定するイベントの条件は開始時間と終了時間のみである。また、JSON 形式のデータで使用するのは、イベントのタイトル、開始時間、終了時間のみなので、jq コマンド^{*10}を使用して必要な箇所のみを取得する。JSON 形式から取得した各講義の時間と、rwho のログイン情報の時間を比較し、その時間中に 1 分でもいた場合に出席として学籍番号を配列に保存する。Google Calendar API のイベント取得の実行例を図 15、図 16 にそれぞれ示す。

```
API_KEY='任意のID'  
CALENDAR_ID='kfucahscpdm1oe7kvs2vhiq16o%40group.calendar.google.com'  
curl -s "https://www.googleapis.com/calendar/v3/calendars/{CALENDAR_ID}/events?key=${API_KEY}"
```

図 15 Google Calendar API のイベント取得の実行例。指定した時間の範囲内のイベントを Curl を用いて API の URL に GET する。Curl コマンドの API_KEY と CALENDAR_ID は任意の ID に変更する。

^{*8} linux の command で、サーバから、もしくはサーバへデータを転送する。

^{*9} テキストベースのデータフォーマット。簡潔に構造化されたデータを記述することができ、記述が容易で人間が理解しやすい。

^{*10} JSON から簡単に値の抜き出し、集計、整形して表示などができる JSON 用の command

```

{
  "kind": "calendar#events",
  "etag": "\"p338ad4kct64ua0g\"",
  "summary": "rwhong",
  "updated": "2019-10-07T09:20:53.298Z",
  "timeZone": "Asia/Tokyo",
  "accessRole": "reader",
  "defaultReminders": [],
  "nextSyncToken": "CNCmkozpieUCEAAYAQ==",
  "items": [
    {
      "kind": "calendar#event",
      "etag": "\"3140792602826000\"",
      "id": "@ba9vaheu4jdgd230i5mk4gtk6",
      "status": "confirmed",
      "htmlLink": "https://www.google.com/calendar/event?
eid=MGJhOXZhaGV1NGpkZ2QyMzBpNW1rNGd0azYgaZ21Y2Foc2NwZG0xb2U3a3ZzMnZoaXExNm9AZw",
      "created": "2019-10-06T21:11:41.000Z",
      "updated": "2019-10-06T21:11:41.413Z",
      "summary": "100966:プログラミング 2",
      "creator": {
        "email": "yuragi0767@gmail.com"
      },
      "organizer": {
        "email": "kfucahscpdm1oe7kvs2vhiq160@group.calendar.google.com",
        "displayName": "rwhong",
        "self": true
      },
      "start": {
        "dateTime": "2019-09-23T12:50:00+09:00"
      },
      "end": {
        "dateTime": "2019-09-23T14:20:00+09:00"
      },
      "iCalUID": "@ba9vaheu4jdgd230i5mk4gtk6@google.com",
      "sequence": 0
    }
  ]
}

```

図 16 JSON の内容。このデータから使用するのは、イベントのタイトル、開始時間、終了時間のみ。この上記の図では、イベントのタイトルが”summary”:"100966:プログラミング 2"を示しており、これが授業名となる。また、後半に書かれている、”start”の”dateTime”が開始時間を、”end”の”dateTime”が終了時間をそれぞれ示している。本システムではこれらの各講義の時間と rwho のログイン情報の時間を比較し、1 分でもその時間中に滞在していた場合、出席と判断し学籍番号を配列に保存する。

7.5 数値化機能

出席確認機能から渡された出席情報を使用して、各学生の評価値を求める。データベースに保存された学生の情報と、出席情報が入った配列とを比較する。比較した結果、出席情報に学籍番号がない場合、欠席と判断しその学生の評価値が加点される。出席情報との比較が終了した時点で評価値はデータベースに保存される。

評価値の算出には、欠席した回数、連続で欠席した回数、何回目に欠席したか、という要素から行う。評価値は式 (5) で求める。

$$X = \sum_{l=1}^m \{a + k(x-1) + bn\}_l \quad (5)$$

ここで、 X は学生の評価値、 a は欠席による加点を表す定数、 k は連続した欠席による加点を求める係数、 x は連続何回目の欠席かを表す変数、 b は講義の重りを表す変数、 n は何回目の講義かを表す変数、 m は欠席回数を表す変数をそれぞれ示す。

数値化のモデルの妥当性を確かめるために、3つのモデルを作成した。

- モデル A

離学トリアージカテゴリーを基に離学者の摘出率を 50% 以上、優良者の誤摘出率を 5% 以下になるようしきい値と加点の上下を調整して作成。

- モデル B

欠席回数を重視したモデル。連続した休みの回数では無く 1 回ごとの欠席を重視し、欠席がまばらな学生に対して摘出率を上げるために作成。

- モデル C

モデル A を基準に摘出タイミングを早くなるようしきい値と加点の上下を調整したモデル。加算値を高め、しきい値を少し高めにして作成。

各モデルの評価値 X の範囲は、モデル A は 0 から 251 を、モデル B は 0 から 176 を、モデル C は 0 から 484 をそれぞれ示している。

7.6 リストアップ機能

数値化機能でデータベースに保存された各学生の点数と、設定している各危険度のしきい値を比較することでリストの作成を行う。点数としきい値を比較した結果は危険度として色で判別し出力される。しきい値の設定には予め作成した色リスト(章6で説明)を基準に行う。

各学生の危険度は一度データベースに保存される。しかし、前回のリストアップ時と同じ危険度だった場合は上書きする必要がないので保存は行われず。また、全ての学生の比較が終了後、データベースに保存された各学生の学籍番号と危険度を出力させる。危険度毎に学籍番号を変数に代入させるが、危険度が更新された学生には学籍番号の右に”_New!!”という文字列を付け加え、ひと目で新規に上がった学生だとわかりやすくする。出力の例として図17に変数の内容を示す。

```
[BLACK]

[RED]

[YELLOW]
s18H4c5f8e60_New!!
s18Hc407420d_New!!

[GREEN]
s18H01ad8fb5
s18H05b37f55
s18H08d1dd65
s18H12d8fe87
s18H198a19b3
s18H1a4ef462
s18H1fcfb289
s18H211020bf
```

図17 リストについて、上から BLACK、RED、YELLOW、GREEN の順に学生の危険度を色別に分け表示。また、新しく危険度が更新された学生は、学籍番号の右に”_New!!”と文字が表示され、ひと目で判別しやすくなっている。

7.7 メール送信機能

リストアップ機能で出力された、危険度別に分けられた学生のリストが入った変数を使い、メールの本文を作成する。メールの Subject を対象の学年、body をリストアップから渡された変数の内容、To を登録されたメールアドレスとし、メールを作成・送信する。実際にシステムから送信されたメールの例を図 18 に示す。

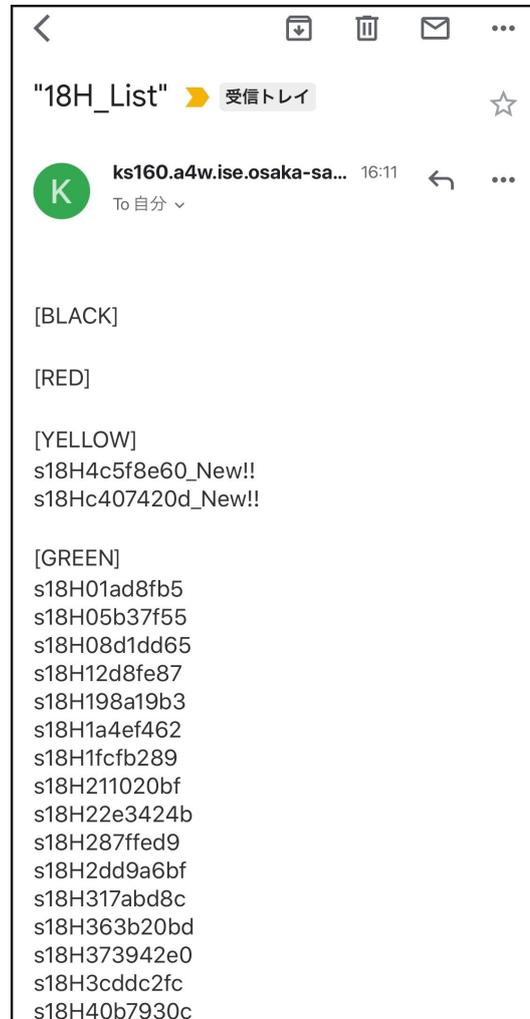


図 18 メールの実際の画面の例。メールの Subject が対象の学年、body がリストアップから渡された変数の内容、To が登録されたメールアドレスを示す。上から BLACK、RED、YELLOW、GREEN の順番に学生を表示。

8 検証

8.1 傾向の数値化

検証には、15H から 18H までの学生の情報を基に作成した数値化モデルを使用し、19H の前期のログイン情報を使用し数値化を行い色リストの作成を行う。評価値の算出は、システムの数値化機能と同様に式 (5) から与えられる。

$$X = \sum_{l=1}^m \{a + k(x-1) + bn\}_l \quad (5)$$

ここで、 X は学生の評価値、 a は欠席による加点を表す定数、 k は連続した欠席による加点を求める係数、 x は連続何回目の欠席かを表す変数、 b は講義の重りを表す変数、 n は何回目の講義かを表す変数、 m は欠席回数を表す変数をそれぞれ示している。

また、作成した数値化のモデルの妥当性を確かめるために、3つのモデルを作成し、比較した。

- モデル A
離学トリアージカテゴリーを基に離学者の摘出率を 50% 以上、優良者の誤摘出率を 5% 以下になるようしきい値と加点の上下を調整して作成。
- モデル B
欠席回数を重視したモデル。連続した休みの回数では無く 1 回ごとの欠席を重視し、欠席がまばらな学生に対して摘出率を上げるために作成。
- モデル C
モデル A を基準に摘出タイミングを早くなるようしきい値と加点の上下を調整したモデル。加算値を高め、しきい値を少し高めにして作成。

各モデルの評価値 X の範囲は、モデル A は 0 から 251 を、モデル B は 0 から 176 を、モデル C は 0 から 484 をそれぞれ示している。また、具体的な数値の設定は表 1 と表 2 にそれぞれ示す。

表 1 各モデルで、学生が休んだときに加算される値の設定。a は欠席による加点を表す定数、k は連続した欠席による加点を求める係数、b は講義の重りを表す変数、n は何回目の講義かを表す変数をそれぞれ示す。

モデル	a	k	bn															
			n	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
A	1	2		3	3	2	1	1	1	2	2	2	2	2	2	1	1	1
B	3	1		3	3	2	1	1	1	2	2	2	2	2	2	1	1	1
C	2	4		6	6	4	1	1	1	2	2	2	2	2	2	1	1	1

表 2 各モデルのしきい値。単位は点。モデル A の各色の範囲は YELLOW は 7 点から 11 点、RED は 12 点から 29 点、BLACK は 30 点以上にわけられる。モデル B とモデル C 各色のしきい値の範囲は YELLOW は 10 点から 19 点、RED は 20 点から 39 点、BLACK は 40 点以上にわけられる。

モデル	YELLOW	RED	BLACK
A	7-11	12-29	30-251
B	10-19	20-39	40-176
C	10-19	20-39	40-484

以下にそれぞれのモデルの詳細と検証結果について説明する。

8.1.1 モデル A

モデル A は一番最初に試作されたモデルで、19H の前期、107 名から検証。授業が 15 回目の時点で離学者の 50% 以上が BLACK・RED に、優良者の 95% 以上が GREEN に分類されるようにしきい値と加点の上下の調整を行った。このモデルの欠点として、15 回目の時点での結果を参考に作成しているの、傾向とのズレがあり各危険度への振り分けがやや遅くなっている。

モデル A を使用した検証結果を図 19 に示す。

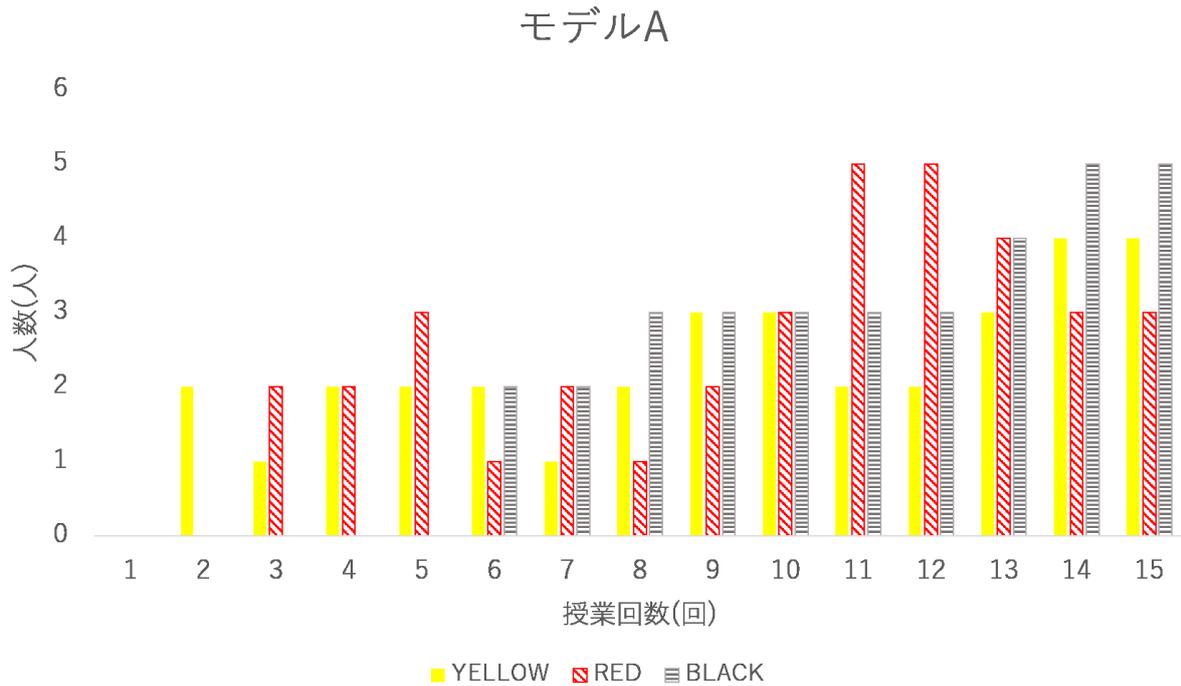


図 19 モデル A の検証結果グラフ。塗りつぶしは YELLOW を、斜線は RED を、横線は BLACK をそれぞれ示している。このモデルでは、15 回目の時点で離学者の 50% 以上が BLACK・RED に、優良者の 95% 以上が GREEN に分類されるようにしきい値と加点の上下の調整を行い作成した。その結果がこのグラフである。縦軸が人数、横軸が授業の回数を示しており、19H の前期、107 名から結果を出している。(モデル B とモデル C も同様のため以下省略する。)

8.1.2 モデル B

モデル B は欠席回数を重視したモデルで、欠席の傾向がまばらな学生でも検出できるようになりモデル A より多くの学生が検出されるようになった。発見した傾向に基づいて作成したモデル A と比較する為に、発見した傾向を重視しない形で作成したのがモデル B になる。

モデル B を使用した検証結果を図 20 に示す。

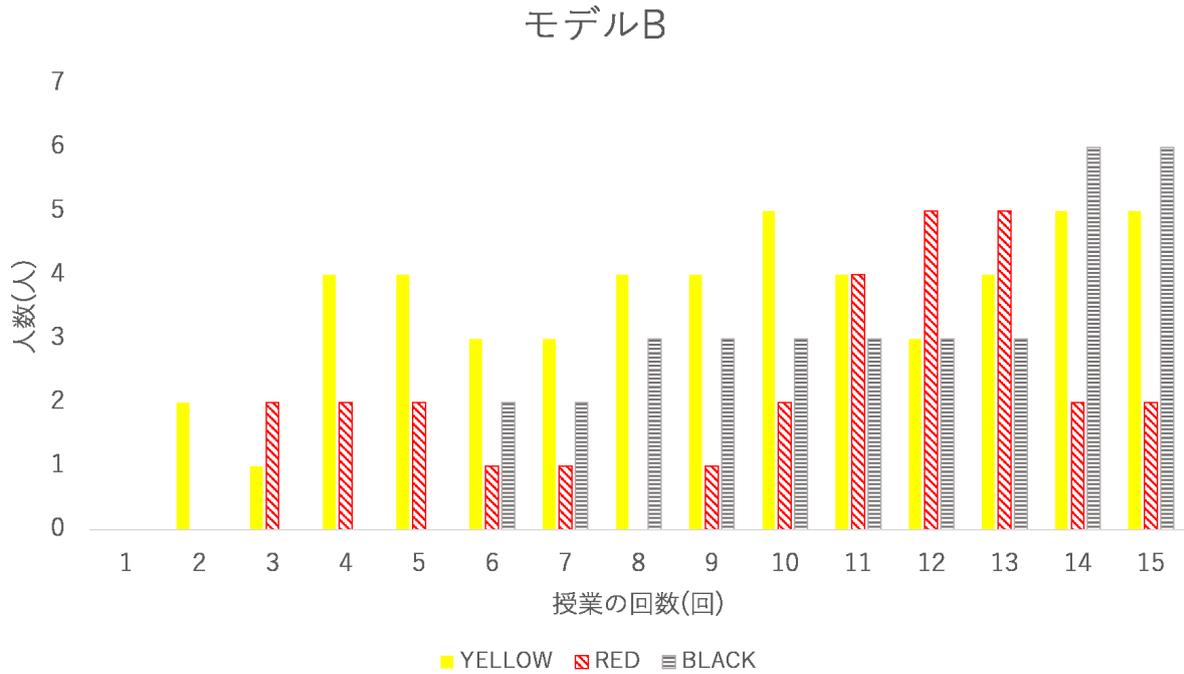


図 20 モデル B の検証結果グラフ。塗り潰しは YELLOW を、斜線は RED を、横線は BRACK をそれぞれ示している。このモデルは欠席回数を重視、欠席の傾向がまばらな学生でも検出できるようになり、モデル A より多くの学生が各色へと検出されるようになった。その結果がこの図である。

8.1.3 モデル C

モデル C はモデル A の欠点だった傾向との誤差を修正したモデルで、加算値としきい値との関係を微調整し、モデル A より早く各危険度への振り分けを行う。設定値は、モデル A から加算値を約 2 倍、しきい値を約 1.5 倍にしている。モデル C を使用して過去のデータを数値化した場合、ほぼ傾向通りの振り分けが可能となった。

モデル C を使用した検証結果を図 21 に示す。

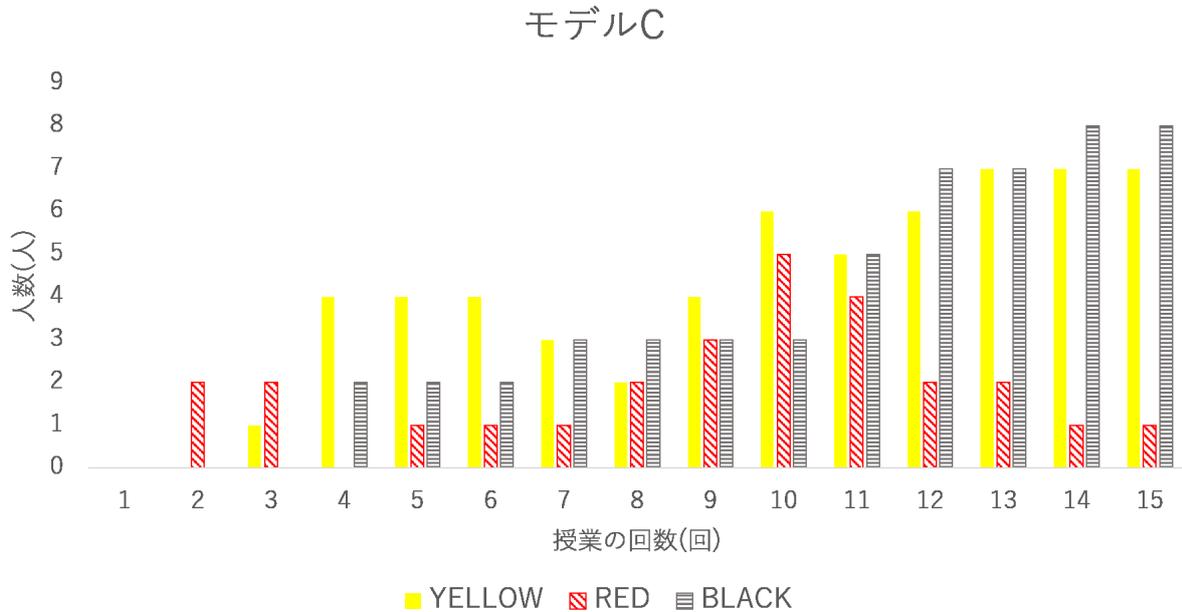


図 21 モデル C の検証結果グラフ。塗り潰しは YELLOW を、斜線は RED を、横線は BRACK をそれぞれ示している。このモデルはモデル A の欠点だった傾向と誤差を修正し、加算値としきい値との関係を微調整したもので、モデル A より早く書く危険度への振り分けを行い、ほぼ傾向通りの振り分けが可能となった。その結果がこの図である。

9 結果と考察

9.1 各モデルの妥当性の考察

19H の 2019 年度前期から出席情報を出し、式 (5) に当てはめて検証を行った結果、各モデルで学生を危険度に分けたものを表 3 に示す。

表 3 19H の 2019 年度前期を対象に検証を行った 15 回目での振り分けの結果。不良者 14 名、優良者 93 名から検出。数値の単位は人。

成績区分	モデル	GREEN	YELLOW	RED	BLACK
不良者	A	6	1	0	7
	B	6	1	2	5
	C	5	2	0	7
優良者	A	89	3	0	1
	B	88	4	0	1
	C	86	5	1	1

どのモデルを使用した場合でも、不良者から離学者になる可能性があるとは判定されたのは 7 人程度で、優良者からは 1 人確認された。

次に、不良者の出席情報と最終危険度の関係をまとめたものを表 4 に示す。

表 4 不良者の出席情報と最終危険度の関係をまとめたもの。授業 15 回目で経験度を各モデルごとに振り分けしている。各回、出席なら 2、遅刻なら 3、欠席なら 0 が表記されている。

学籍番号	危険度			出席情報														
	A	B	C	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
s19H09854fba	B	R	B	2	2	2	2	2	2	2	2	0	0	0	0	2	2	2
s19H3a450f82	G	G	G	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
s19H3a7838f2	G	G	G	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
s19H3dbd5e46	B	R	B	2	2	2	0	0	2	2	0	0	0	2	2	0	0	0
s19H5c1ae998	G	G	Y	2	2	2	2	2	2	2	2	0	0	2	2	2	2	2
s19H69a59eae	G	G	G	2	2	2	2	2	2	2	2	2	2	2	0	2	0	2
s19H7c334a24	G	G	G	2	2	2	2	2	2	2	0	2	2	2	2	0	2	2
s19H80dc0e11	G	G	G	2	2	2	0	2	2	2	2	2	2	2	2	2	2	2
s19Hbeb3d037	Y	Y	Y	2	0	2	0	2	2	2	2	0	2	0	2	2	2	2
s19Hc7bf0e4f	B	B	B	2	2	2	2	2	2	2	0	0	0	0	0	0	0	0
s19Hd41525a7	B	B	B	2	2	0	2	2	0	2	2	0	0	0	0	0	0	0
s19Hd438d1f9	B	B	B	2	2	0	0	0	0	0	0	0	0	0	0	0	0	0
s19He366632f	B	B	B	2	0	2	0	0	2	0	2	0	0	0	2	0	0	0
s19Hfbe2f690	B	B	B	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

モデル毎の最終危険度を比べると、モデル A とモデル C は一人を除いて同じ判定結果が出ており、概ね傾向通りである。それに対し、モデル B を傾向に乗っ取って判定すると他のモデルでは BLACK と判定されていた学生が RED と判定されている。これは、BLACK のしきい値の高さと、連続欠席回数による傾向が十分に取り入れら

れていない為だと考えられる。この時点でモデル B は正確性に掛けてると判断したため、離学者を発見するには十分で無いと結論を出した。

ここからはモデル A とモデル C を比較するために、BLACK と判定された学生を傾向別に分け、どのタイミングでどの危険度に判定されているのかを確認する。本検証では、最終危険度が BLACK と判定された学生は大きく 3 つに分けることができる。

- 最初から一度も出席していない学生
- 途中から一度も出席しなくなった学生
- 所々欠席があり、最終的に出席しなくなった学生

これらの特徴が挙げられる学生がどのタイミングで、どの危険度に判定されたかをモデル A、モデル C で比較することで、どちらのモデルがより妥当かを判断する。

9.1.1 最初から一度も出席していない学生

最初から一度も出席していない学生の例として表 5 の学生を確認していく。

表 5 例:最初から一度も出席していない学生。各回、出席なら 2、遅刻なら 3、欠席なら 0 が表記されている。

学籍番号	授業回数														
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
s19Hfbe2f690	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

この学生のモデル A、モデル C の各危険度の判定タイミングを表 6 に示す。

表 6 モデル A、モデル C の危険度の判定タイミング。数値は何回目の授業で判定が出されたのかを示しているため、単位は回。

モデル	YELLOW	RED	BLACK
A	2	3	5
C	なし	2	4

傾向を基に作成した離学トリージカテゴリーと比較すると、モデル A はどの色も検出タイミングが 1 つ遅く、モデル C は離学トリージカテゴリー通りのタイミングで検出している。

9.1.2 途中から一度も出席しなくなった学生

途中から一度も出席しなくなった学生の例として表 7 の学生を確認していく。

表 7 例: 途中から一度も欠席しなくなった学生。各回、出席なら 2、遅刻なら 3、欠席なら 0 が表記されている。

学籍番号	授業回数														
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
s19Hc7bf0e4f	2	2	2	2	2	2	2	0	0	0	0	0	0	0	0

この学生のモデル A、モデル C の各危険度の判定タイミングを表 8 に示す。

モデル A は 2 回連続の欠席で YELLOW と判定できず、3 回目の欠席でいきなり RED の判定をだしている。その半面モデル C は離学トリージカテゴリー通りのタイミングで検出が行われている。

表 8 モデル A とモデル C の危険度の判定タイミング。数値は何回目の授業で判定が出されたのかを示しているため、単位は回。

モデル	YELLOW	RED	BLACK
A	なし	10	12
C	9	10	11

9.1.3 所々欠席があり、最終的に出席しなくなった学生

所々欠席があり、最終的に出席しなくなった学生の例として表 9 の学生を確認していく。

表 9 例:所々欠席があり、最終的に出席しなくなった学生の例。各回、出席なら 2、遅刻なら 1、欠席なら 0 が表記されている。

学籍番号	授業回数														
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
s19He366632f	2	0	2	0	0	2	0	2	0	0	0	2	0	0	0

この学生のモデル A、モデル C の各危険度の判定タイミングを表 10 に示す。

表 10 モデル A とモデル C の危険度の判定タイミング。数値は何回目の授業で判定が出されたのかを示しているため、単位は回。

モデル	YELLOW	RED	BLACK
A	5	7	14
C	4	7	11

モデル A は RED まで離学トリージカテゴリー通りのタイミングで検出しているが、BLACK と判定されたのが 14 回目とかなり遅くなっている。モデル C は YELLOW と判定したタイミングがやや早く、BLACK への判定もモデル A より早い。しかし、離学トリージカテゴリーの基準では遅い結果となった。

9.1.4 結論

比較した結果、モデル A よりモデル C の方が離学トリージカテゴリーに則った判定ができています。しかし、両方のモデルとも来たり来なかったりといった欠席の仕方をしている学生が 4 人おり、それに対して RED までの判定はそれなりの精度があるが、BLACK への判定の精度が低いと言える。なぜなら、本研究は離学者になる可能性のある学生を早期発見することが目的の為である。しかし、トータルで見ると RED までの判定の精度が高ければ離学者の早期発見に繋がると言えるので、モデル C は十分に離学者を早期の段階で発見できる。

10 まとめ

本研究では、本学の離学者の数が問題となっていることから、rwho の情報を用いて過去 4 年分のデータから傾向を出し、離学者になるであろう学生を自動判断したメールを教員に通知する離学者トリアージシステムを開発した。

19H の学生の前期のデータのみで検証を行ったため断定はできないが、検証するために作った 3 つのモデルのうち、判定の精度が高かったモデル C を使用することで、学生が色別・危険度別に振り分けられる。

その結果、このシステムを使用することで、最優先に対策しなければいけない学生が明確にわかり、教員の支援と早めの対策ができると考えている。

今後は本番環境で実際に教員に使用していただき、使っていただいたフィードバックでシステムの精度を上げ、離学者が減少していくかを検証していかなければならない。

10.1 今後の課題

今後の課題として、今回は rwho の情報と講義の出欠を主に傾向を出していたが、遅延や雨天による欠席・席の座る場所・友達関係なども本研究に取り入れ、新たな傾向をだしていく。出した各モデルの学生の評価値 X の範囲がバラバラなため、モデルの正規化を行う。新入生でのモデルの再検証。また、結果を元に実際に教員に使っていただき、離学者が減少するかの検証。以上が今後の課題である。

謝辞

本論文執筆及び研究作業等、研究室での活動の際以外にもご指導・ご協力いただきました大垣 斉准教授、情報教育システム研究室所属の学生、当研究室のOBに深く感謝いたします。

参考文献

- [1] 篠原拓哉. 災害時のトリアージの現状-救急医療の現状と課題 (後編). 2016.

付録 A ソースコード

A.1 数値化プログラム

Listing 1 foo

```
#!/bin/bash
. ./mymysql.sh
. ./sign_up.sh
. ./quantification.sh
. ./color_up.sh
. ./list_up.sh

#のバックアップIFS
IFS_BUCK=$IFS
#を使用するための変数Google_Calendar
APIKEY='AIzaSyAfzwwshNT9REZJOTTn5fbqDjLbq99RCxzA'
CALENDAR_ID='kfucahscpdm1oe7kvs2vhiq16o%40group.calendar.google.com'

#各種日付関係
MAXDATE='date +%Y-%m-%d'
NOW_m='date +%m'
if [ 4 -le ${NOW_m} -a ${NOW_m} -le 8 ]
then
    MIN_DATE='date +%Y-04-01'
    PERIOD='date +%YA'
elif [ 9 -le ${NOW_m} -a ${NOW_m} -le 12 -o 1 -le ${NOW_m} -a ${NOW_m} -le 2 ]
then
    MIN_DATE='date +%Y-09-01'
    PERIOD='date +%YB'
fi
MIN_TIME="' ${MIN_DATE}T08%3A00%3A00.000%2B09%3A00'"
MAX_TIME="' ${MAXDATE}T22%3A00%3A01.000%2B09%3A00'"

#の取得するGoogle_CalendarJSONURL
URL="https://www.googleapis.com/calendar/v3/calendars/${CALENDAR_ID}/events?timeMax=${MAX}"
#URL="https://www.googleapis.com/calendar/v3/calendars/${CALENDAR_ID}/events?key=${APIKEY}"

JSON='curl -s "${URL}"'

#イベント数
```

```

JSONLENGTH='echo $JSON | jq '.items' | jq length '

#イベント配列
declare -a LECTURES=()

#イベントを取得して配列に格納
for i in `seq 0 $(expr ${JSONLENGTH} - 1)`
do
    SUMMARY='echo $JSON | jq -r .items[${i}].summary '
    START='echo $JSON | jq -r .items[${i}].start.dateTime '
    END='echo $JSON | jq -r .items[${i}].end.dateTime ' イベント名

    # 開始時間終了時間
    LECTURES[${i}]='echo ${SUMMARY} ${START} ${END} '
done

#イベント名から講義の種類を取得
for i in `echo $JSON | jq -r .items[].summary | sed -e "s/:.*/g" | uniq `
do講義の対象年次を取得
    #
    snedMySQLCommand "SELECT is_year FROM lectures_informations WHERE lecture_id=${i};"
    TARGET='echo $(expr $(date +%Y) - $ret + 1)"H" | cut -c 3-'出席者配列

    #
    declare -a ATTENDANCE=()

    IFS=$'\n'行われた講義から出席者を取得

    #
    for j in `echo "${LECTURES[*]}" | sort | grep $i `
    do
        IFS=$IFS_BUCK

        START='date +%Y-%m-%d_%H:%M:%S' -d $(echo $j | awk -F'_' '{print $2} ' ) '
        END='date +%Y-%m-%d_%H:%M:%S' -d $(echo $j | awk -F'_' '{print $3} ' ) '
        #echo $START $END

        snedMySQLCommand "SELECT DISTINCT student_id FROM rlasts WHERE '${START}' <= login AND

    IFS=$'\n'

```

```

ATTENDANCE="( "${ATTENDANCE[@]} " $ret )
done

IFS=$IFS_BUCK新規登録

#
Sign-Up ${PERIOD} ${TARGET}数値化関数

#
Quantification ATTENDANCE ${PERIOD} ${TARGET}カラーアップ関数

#
Color-Up ${PERIOD} ${TARGET}リストアップ関数

#
List-Up ${PERIOD} ${TARGET}
done

```

A.2 メール送信プログラム

Listing 2 hoge

```

#!/bin/bash

#を通すPATH
export PATH=$PATH:/usr/sbin

MAIL_TO="s16h010@ge.osaka-sandai.ac.jp"

MAIL_FROM="ks160.a4w.ise.osaka-sandai.ac.jp"

SUBJECT="メールテスト"

DATA="データ"

mail_send () {

cat << EOD | nkf -j -m0| sendmail -t
From: ${MAIL_FROM}
To: ${MAIL_TO}
Subject: "${2}_List"

```

MIME-Version: 1.0

Content-Type: text/plain; charset="ISO-2022-JP"

Content-Transfer-Encoding: 7bit

'echo -e \${1}'

EOD

}