

オンライン数式入力演習・評価システム Möbius を利用した 大学初年次線形代数科目の実践

龍谷大学先端理工学部 樋口三郎¹

1 はじめに

著者は 2022 年度前期に、数学系の大学初年次学生向けの線形代数の授業を担当した。この時期には、新型コロナウイルス感染症による授業実施への制限があったため、対応してオンラインツール、特にオンライン数式入力演習・評価システム Möbius[1] を導入した授業をデザイン・実施した。この授業について報告する。

第 2 章では、科目の実施に課された制約から選択した授業デザインについて述べる。第 3 章では、オンライン数式入力演習・評価システム Möbius の利用について述べる。第 4 章では、アンケートや成績データなどに基づく評価を述べ、第 5 章で結論を述べる。

2 授業デザイン

報告する科目「線形代数及び演習 I」は龍谷大学先端理工学部の科目で、数理・情報科学課程の 1 年次前期の必修科目である。高等学校の数学 B のベクトルを前提とし、実数ベクトルを対象として、行列の計算、連立 1 次方程式、行列式を扱い、抽象的な線形空間論は最後にごく一部だけ触れる。これは選定した教科書 [2] の第 0-4 章と第 5 章の一部に相当する。

2.1 授業実施の条件

履修登録者数は、1 年生 113 名と再履修の 18 名だった。教員 1 名 (筆者)、および大学院生のティーチングアシスタント 6 名が担当した。

2022 年度は新型コロナウイルス感染症の影響が残る年度だったが、大学の決定により、この科目は、教室内で学生が離れて着席するなどの対策を行った上で、対面授業として実施することが求められた。前期の 15 週の水、金曜日の 1 講時が割り当てられた。

また、大学の、新型コロナウイルス感染症に対する明示された方針と、過去の方針適用の実例から、社会の感染状況が急に変化すると、短い予告で対面授業とオンライン授業との間の切替が起こることが予想された。また、大学の方針として、感染者や濃厚接触者になったことによる欠席は不利に扱わないことが求められた。さらに、教員の間では、感染の可能性のある学生が成績を気にして登校することを避けるべき、という認識が共有されていた。

¹E-mail: hig@math.ryukoku.ac.jp

2.2 感染・オンライン授業への切替の可能性への対応

対面授業とオンライン授業との間の切替が起きたときに、授業内の各活動は、後述するように、そのまま移行できることを考慮した。また、対面授業の時期に感染や濃厚接触した学生が学習に大きな障害を持たないようにすることを考慮した。

オンライン授業下での非参照非相談の試験の実施が難しいことはよく知られている。期末試験ですべての評価を行うデザインにした場合、期末試験直前にオンライン授業への切替が起きると、正確な評価が行えなくなる可能性がある。また、授業全体がオンライン授業に切り替わらないまでも、一部の学生や教員が感染や濃厚接触などの理由で期末試験に出席できない可能性は大きい。これらの事情を考えると、評価はなるべく分散することが望ましいと結論した。具体的には、ある回の授業で初めて出てきた事項に対して次の回の授業の最初に行う 10 分程度の紙の小テストを主要な評価 (90%) とした。

一方で、2022 年度は、大学全体を見ると、オンライン授業が減って対面授業が増える時期であり、学生からは各授業について「この内容はなぜわざわざ対面にする必要があるのか」という問がよく発された。このことから、前述と逆の要求、すなわち対面授業ではオンライン授業にない価値を付与する必要があると考え、後述するチーム課題 (2.5 節) を導入した。

2.3 授業のデザイン

システム Möbius (3 章) に加え、ファイルの配布や提出の機能を持つ学習管理システム (LMS) として Moodle を、ビデオ会議やテキストチャットの機能を持つコミュニケーションツールとして Microsoft Teams を使用した。

Möbius は LMS からリンクはするが、それ以外は独立に運用する。Möbius 自身も、ファイルの配布や提出、学生への通知などの LMS としての機能を持つが、この授業ではオンライン数式入力演習・評価システムの機能のみを使用した。

30 回の授業に講義と演習のような区別は行わず、対面の各回の授業と、次回の授業までの期間に、次の順序で各活動を行うことにした。

小テスト (授業の最初 10 分) 学生が小テストを受験する

説明 (授業中) 教員が配布資料をプロジェクターと画面共有で提示する説明

例題 (授業中) 教員が配布資料をプロジェクターと画面共有で提示する説明

Möbius (授業数回に 1 回) 学生が指定された Möbius の問題を解く

チーム課題 (授業の最後 20 分程度) 学生が一定の制限下で自由に 2 人組または 3 人組を作り、ランダムに割り当てられた問題を会話しながら共同で解く

小テスト予告 教員が、次回の小テストでは何ができることをチェックするかを明示する

Möbius (授業時間外) 学生が指定された Möbius の問題を解く

フィードバック (授業時間外) 教員が小テストとチーム課題の採点添削済み答案を教員がスキャンしてファイルとして LMS で返却する

予告 (授業時間外) 教員が次回の授業と小テストの予定と、チーム課題を採点して判明した注意点を、メールと Microsoft Teams で送る

学生はノート PC を持参することになっており、教室で Microsoft Teams のビデオ会議に参加する。資料提示は画面共有できる範囲で行い、教員の音声とともに録画する。

学習のゴールは授業で学んだ内容を理解しているかを測る小テストである。授業で内容を説明した後、小テストの出題予定に近い問題や、その一部分に相当する問題を、例題、チーム課題、Möbius で繰り返し出題する。学生が理解していない場合にはチーム課題で理解を進めること、誤解している場合にはチーム課題の採点結果または Möbius の自動採点結果により気づくこと、Möbius で繰り返し練習できること、授業内の対面での質問に加えて、授業時間外でもメールや Microsoft Teams で質問できることを意図している。

講義と、配布した問題で学生が自由に行う演習を組み合わせた形態の授業と比べると、演習に相当する部分の活動を細かく制御しており、自由な演習は Möbius の問題として授業時間外に置いていると言える。

オンライン授業に切り替えた場合は次のように置き換える。配布資料は、対面では紙とファイルの両方で配布しているが、ファイルのみとする。小テストは、対面でも使用している Möbius 上で行う。説明と例題は、同期オンライン授業なら、対面と同様にビデオ会議で行う。非同期オンライン授業ならオンデマンドの動画で行う。チーム課題は行わない。紙での提出物がないため、これに対するフィードバックは不要になる。予告と授業外での質問対応は同様に行う。

2.4 オンライン授業移行リハーサル

初年次学生の大学への適応が進んだ 11 回目、5 月初旬に、1 回のオンライン授業を設定した。これは、それ以降に求められるかもしれない対面授業からオンライン授業への移行のリハーサルで、参加準備での疑問や、参加に失敗したときの学習の補充が対面でできる状態で試行しておくという意図である。

2.5 チーム課題

チーム課題は、自分の理解を他者に話したり、他者の説明を聞いて正誤を判定したりする活動の材料となる課題である。このような共同作業に適した問題がどのようなものかについて、現時点では明確な実験結果や合意はないと考える。この実践では次のようなことを考慮している。

まず、授業時間内の短時間で多くの学生が完成することを想定しているため、所要時間が短くばらつきが小さいものを選定している。学んだ手順を実行すると完成するもののほうがこの基準にあてはまりやすい。一方、大量の試行錯誤が必要な問題は時間を要し、「思いつく」ことが必要な問題は時間がばらつき、この基準にあてはまりにくい。

次に、1 年生の段階では自分の考えを整理して述べることに慣れていない学生が多いため、答案上に各自の考えが一定のペースで現れ、相手に伝わるような問題を選定している。学んだ手順を実行するような課題は結果が一定のペースで現れやすく、図を描く問題は相手に伝わりやすいため、この基準にあてはまりやすいと考えている。一方、思いつきに基

づいて不可分な全体を完成させて論理をチェックするような証明問題は、途中段階で相手に伝える活動が起こりにくいと想定し、出題には慎重になっている。

オンライン授業の場合にチーム課題を行う方法を検討したが、この授業では、行うことは計画せず、リハーサル (2.4 節) でも実施しなかった。

数学の答案作成過程で少人数で議論を行うという観点からは、同期オンライン授業の場合は、ビデオ会議のブレイクアウトルーム機能でランダムな 2 人組を作って共同で答案の文書を作成する、非同期オンライン授業の場合は、LMS 上で各自の答案の相互採点や掲示板での議論を行う、などの方法が考えられた。これらの方法が数学の学習においてどの程度効果的かは十分に調べられていない。

これらの方法で想定される困難は次のようなものである。まず、数学やオンラインの作業に限らず、文書の共有に端末を用いた場合の議論は紙を用いた場合よりも認知負荷が高いことが知られている [3]。数式を含む文書を共有するのに、数式を記述可能なワードプロセッサを用いる場合、不慣れな学生の場合手書きよりも遅くなる傾向がある。手書き文書を共有するには、タッチペンやカメラを用意するなど、より高度な環境の準備が必要になる。また、オンラインのグループワークでは共有できる文書の個数が限定されるため、授業資料、教科書などの複数の文書の参照箇所を伝え合いながら作業を行うことにも困難があると考えられる。

3 Möbius の使用

3.1 オンライン数式入力演習・評価システム Möbius

Möbius[1] は数学 e ラーニングシステム [4] のひとつであり、システムは、数値をランダムに変更した問題を出題し、学生の数式による解答をアルゴリズムにしたがって採点する機能を持つ。クラウドサービスとして有料で提供され、教員・学生は Web 上で作問・解答を行う。クラウドのバックエンドでは、出題・採点に数式処理システム Maple が使用されており、以前は Maple T.A. とよばれていた。

Maple T.A. の開発当初から、カナダ、ヨーロッパの複数の大学での使用例が報告されている [5, 6, 7, 8]。日本国内の大学教育においては、先駆的な北本の使用例 [9] や、和久井によるオンライン授業での使用例 [10] がある。

数学 e ラーニングシステムでよく知られたものとして、数式処理システム Maxima に基づく STACK[11] がある。STACK と Möbius の基本的なデザインは共通で、バックエンドの数式処理システムが問題を生成し、Web 上で入出力を行い、学生の解答入力を数式処理システム上のオブジェクトと見なして、数学的等価性に基づいて採点する。

STACK は、オープンソースの学習管理システム (LMS) である Moodle のプラグイン (小テストの問題タイプ) として提供されており、Moodle の小テストと問題バンクの仕組みを用いて、小テストと問題を組織することができる。これに対し、Möbius は自身が基本的な LMS の機能をもっていて、課題とコンテンツレポジトリの仕組みで、課題と問題を組織することができる。STACK が、学生の入力した解答に基づいて点数に留まらないフィードバックを返すことができるという特徴を持つのに対し、Maple は、数式を含む多様な解答方式を持つという特徴がある。

一般に、オンライン数式入力演習・評価システムを授業で使用すると、教員は学生の答案採点の労力を減らすことができる。一方、作問にはより多くの労力が必要である。自動採点を行う問題はプログラムであり、学生のどのような解答入力にも適切な点数を返すことを保証する必要がある。さらに、ランダム出題を行う場合には、単なる1問題を作るのではなく、乱数をパラメタとする1問題族を作る必要がある。この問題族は、すべて数学的に意味のある問題である必要があり、使用する状況によっては、難易度が大きく違わないことが求められる。問題作成については3.4節で述べる。

また、オンライン数式入力演習・評価システムは、学生が、手書きでなく、一定の文法に従った解答入力をする必要があるため、学生が解答入力に習熟できるように教員が配慮する必要がある。

3.2 対象授業への Möbius の導入

龍谷大学理工学部では、2015年度から初年次学生への学部共通数学教育 [12]、推薦入学による入学予定者への入学前教育 [13] に Möbius を利用してきた。

Möbius は 2022 年度も初年次学生の学部共通数学教育に利用されることになっており、他の科目でも利用可能だった。また、初年次の学生は学部共通数学教育を通じて Möbius の数式入力に習熟することが期待された。

この状況を利用し、この科目の演習を Möbius 上で行うことにした。Möbius では過程の記述を評価することはできないが、学生が何回でも同種の問題を取得して自動採点結果を確認できることから、特に計算技能の練習には適している。過程の記述は、チーム課題 (2.5 節) や少数のレポート課題 (3.5 節) を通じて学習することにした。最終的に、121 問 (重複出題を含めれば 140 問) の問題を 58 個の課題に分けて出題し、課題を繰り返し受験して、最高点を満点にすることを求めた。全問題の最高点の合計を、成績の 10% を占める平常点のさらに一部として算入することにした。また、期限は定めず、満点が得られた後も、必要に応じて練習することを薦めた。

3.3 Möbius の問題の出題と解答方法

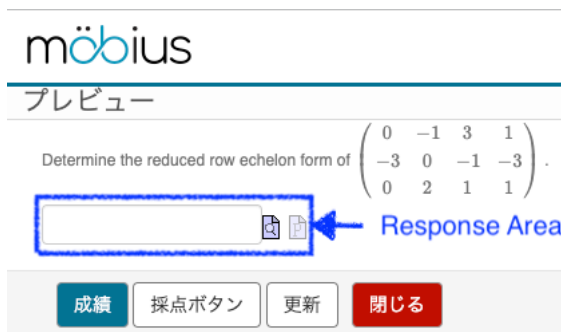
学生が Möbius の演習に取り組む際には、教員の指示により、クラウド上の Möbius の、特定の大学の特定のクラスを訪れる。クラス内には、演習問題が、課題-問題の階層に配置されている。この中から特定の問題に取り組むことになる。

学生が課題を選ぶと、問題が順に表示される (図 1a)。学生は、Response Area 欄に、特定の文法で解答を入力する。

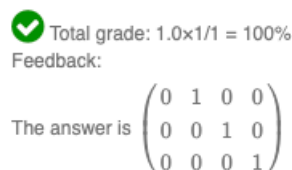
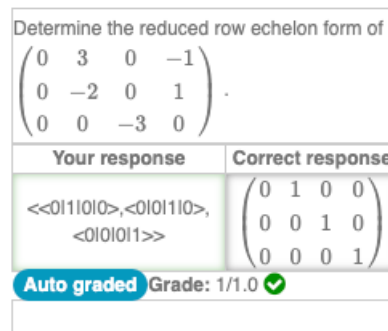
学生は、教員による課題の設定により、直ちに、または締切後に自動採点結果 (図 1b) を得る。

3.4 Möbius での線形代数の問題の実装

線形代数の行列やベクトルを Möbius で扱う場合、Maple の LinearAlgebra パッケージを使う方法、linalg パッケージを扱う方法、リストなどを利用して独自に実装する方法が考えられるが、ここでは最も新しい LinearAlgebra パッケージを採用した。



(a) 問題の例



(b) 自動採点結果の例

図 1: Möbius の問題と自動採点結果の学生への表示

ランダムに生成した行列を学生に表示し、その簡約階段形の解答を求めるといった典型的な問題の作成例を挙げる。

教員用の問題作成ページ (Question Designer) では図 2 のように表示される。Question Text 欄 が学生に表示される問題文、Feedback 欄が解答後に学生に表示される解説である。\$A\$disp, \$ans\$disp の部分は、Algorithm 欄に書かれた Möbius 独自のコードで定義されたそれぞれの値 (Algorithm Preview 欄の Value の列) と置換される。この問題は学生に対して図 1a のように表示される。

3.4.1 問題生成アルゴリズム

図 2 の Algorithm 欄で、maple("命令") はバックエンドの Maple の命令の呼び出しを示す。

第 1 行では、LinearAlgebra パッケージの RandomMatrix 関数を用いて、出題する行列をランダムに生成し、Möbius の変数 \$A\$ に代入している。第 2 行では、\$A\$ の簡約階段化を行い、結果を \$ans\$ に代入している。第 3 行では、Maple のオブジェクトを Web ページで表示できる MathML 形式に変換するユーザ定義関数 \$disp\$ を定義している。第 4,5 行では、関数 \$disp\$ により、\$A, ans\$ をそれぞれ変換して、表示用の変数 \$A\$disp, \$ans\$disp に代入している。

任意の行列に簡約階段形があることから、この手順によって、ただ一つの正解を持つ数学的に正しい問題が生成されるが、学習用の問題としては、要する基本変形が長い、その途中で複雑な分数が現れる、など、簡約階段化とは別の困難が現れてしまうことがある。そのような別の困難を避けたいこともあるだろう。

そのような場合には、簡約階段行列にランダムにいくつかの行基本操作を施すことで行列 \$A\$ を生成すれば、問題と答がともに簡単な行列で、簡約階段化に要する基本変形が短い

- Question Text

- Algorithm

```

1  $A=maple("randomize(): LinearAlgebra[RandomMatrix](3,4,density=0.5,generator=-3..3)");
2  $ans=maple("LinearAlgebra[ReducedRowEchelonForm]($A)");
3  $disp=maple("proc(A) printf(MathML[ExportPresentation](A)) end proc;");
4  $Adisp=maple("($disp)($A)");
5  $ansdisp=maple("($disp)($ans)");

```

Algorithm Preview		
Variable	Value	Range
A	Matrix(3,4,{(1, 2) = 2, (2, 1) = 2, (2, 2) = -2, (2, 4) = 2},datatype = anything,storage = rectangular,order = Fortran_order,shape = [])	
ans	Matrix(3,4,{(1, 1) = 1, (1, 4) = 1, (2, 2) = 1},datatype = anything,storage = rectangular,order = Fortran_order,shape = [])	
disp	proc (A) printf(MathML[ExportPresentation](A)) end proc	
Adisp	$\begin{pmatrix} 0 & 2 & 0 & 0 \\ 2 & -2 & 0 & 2 \\ 0 & 0 & 0 & 0 \end{pmatrix}$	
ansdisp	$\begin{pmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$	

- Feedback

図 2: Möbius の教員用の問題作成ページ (一部を削除)

表 1: Maple の LinearAlgebra パッケージにおける行列・ベクトルのテキスト入力方法

種類	入力する値	入力テキスト
縦ベクトル	$\begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}$	<1,2,3>
横ベクトル	[1 2 3]	<1 2 3>
行列	$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix}$	<<1 2 3>,<4 5 6>>または<<1,4> <2,5> <3,6>>

ものを得ることができる (付録 A). 多くの問題では, このように, 単純な答をランダムに生成し, そこから問題を作成するようにしている.

3.4.2 解答入力欄・正誤判定の設定

問題作成ページ (図 2) において, 解答入力欄は Question Text 欄内の Response Area と注記された部分であり, 問題ページ (図 1a) では Response Area と注記されたテキストボックスとして表示される.

問題作成ページにおいて, Response Area の部分をクリックすると, 入力のタイプや, 正誤判定のコードを記述するポップアップが現れる (図 3a).

ポップアップの Expression Type, Text/Symbolic entry 欄で, 学生の数式入力方法を指定する. Möbius では, 学生は数式エディタやテキストで数式を入力できるが, この授業では, 学部共通教育の課題にあわせて, 行列やベクトルの入力は LinearAlgebra パッケージの Maple の文法 (表 1) にしたがって, テキストで入力する方法に統一した.

ポップアップ内では, 学生の入力を採点し, 正解に true, 不正解に false を返すような Maple の採点コードを Grading Code 欄に指定する.

問題新規作成後のデフォルトでは, 学生の入力の実数や実数値関数であることを想定して,

```
evalb(($ANSWER)-($RESPONSE)=0);
```

というコードが Grading Code 欄に入力済になっている. しかし, LinearAlgebra ライブラリの Matrix オブジェクトである正解と学生の入力とが等しいかを判定するには, 等号でなく図 3a の Grading Code 欄の採点コードのように LinearAlgebra ライブラリの Equal 関数で比較する必要がある.

図 3a の Grading Code 欄の採点コードは正しい採点を行うが, 2つの欠点を持つ.

ひとつは, 図 3b の Correct response 欄の表示のように, 表示される正解が Maple の LinearAlgebra パッケージの Matrix オブジェクトの内部表現であり, 学生にとって理解が難しいことである (図 3b).

もうひとつは, Grading Code 欄に書く採点コードが, 問題作成ページの Algorithm 欄に書くコードと深く関係する内容でありながら, クリックしないと開かない離れたポップアップ内の Grading Code 欄に置かれ, 教員にとって両方を通覧してのチェックや修正が困難なことである.

これらの欠点を解消するために図 4 のような実装を行っている.

ひとつめの欠点に対しては, 図 4 の Answer 欄を $\$disp(\$ans)$ とすることで, 図 1b の Correct responses 欄のように, 正解の行列が MathML による数式として表示される.

Edit Response Area

Question Feedback

Maple-graded:

Weighting: 1

Answer: \$ans
(referenced when grading as \$ANSWER)

Grading Code: LinearAlgebra[Equal](\$ANSWER,\$RESPONSE);

Expression Type: Maple Syntax - e.g. diff(2*f(x),x) v

Text/Symbolic entry: Text entry only v

Determine the reduced row echelon form of $\begin{pmatrix} -3 & 3 & 0 & 0 \\ 3 & 1 & -3 & -3 \\ -2 & 0 & 0 & 0 \end{pmatrix}$.

Your response	Correct response
<1,2><2,3>	Matrix(3,4,[[1, 1], [2, 2], [3, 3]]) = 1, (2, 2) = 1, (3, 3) = 1, (3, 4) = 1, datatype = anything, storage = rectangular, order = Fortran, order, shape = []

Auto graded Grade: 0/1.0

Total grade: 0.0 x 1/1 = 0%

Feedback:

The answer is $\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{pmatrix}$

(a) ライブラリ関数 Equal による正解判定

(b) Matrix オブジェクトの内部表現

図 3: Response Area 欄の設定例 1 とその正解表示

Edit Response Area

Question Feedback

Maple-graded:

Weighting: 1

Answer: \$disp(\$ans)
(referenced when grading as \$ANSWER)

Grading Code: (\$grade)(\$ans,\$RESPONSE);

Expression Type: Maple Syntax - e.g. diff(2*f(x),x) v

Text/Symbolic entry: Text entry only v

Optional:

Maple Repository: Maple Repository

Plotting Code:

Custom Previewing Code:

図 4: Response Area 欄の設定例 2

もうひとつの欠点に対しては、問題作成ページ (図2) の Algorithm 欄で、正解判定のためのユーザ定義関数 `$grade` を例えば

```
$grade:=maple("proc(A,R) LinearAlgebra[Equal](A,R) end proc;");
```

のように定義し、これを図4の Grading Code 欄に

```
($grade)($ans,($RESPONSE));
```

のように指定することで、通覧が可能になる。

このように正解判定の関数を記述する場合には関数 `$grade` はかなり複雑なものになりうる。これを記述・修正・テストするときには、問題のプレビューを行い、実際に回答を入力してみる必要があり、時間と労力を要する。これを避ける方法として、図2の Algorithm 欄内にテストコードを例えば次のように書いておくことができる。

```
$test1:=maple("($grade)($ans,$ans)");
```

```
$test2:=maple("not ($grade)($ans,Matrix(1..3,1..4,-1))");
```

`$grade` が正しく定義されていれば、これらの値は `true` となるはずで、それは、図2の Algorithm 欄の編集集中に Preview Algorithm 欄で常時確認できる。

以上の工夫を行ったコードの例を付録Aに示す。

もちろん、このテストをパスしても、ポップアップ (図4) 内の Grading Code 欄に関数 `$grade` が正しく記述されている保証はない。しかし、この欄はいったん入力した後で編集することは少なく、問題とはならない。

3.5 Möbius の様々な用途による課題タイプの選択

毎回の授業後の演習には、課題タイプとしてマスターリー課題を使用した。これは、ある問題セットに属する問題が繰り返し出題・採点され、すべての問題の最高点が満点となるまで続けるものである。数値は出題のたびにランダムに決定される。

また、オンライン授業のリハーサルの際の小テストや、感染や濃厚接触による通常の回の小テスト欠席の代替となる小テストには、課題タイプとして、小テスト/宿題を使用した。一定の制限時間、受験回数が科され、問題の採点結果が表示されないものである。

Möbius を用いて、学生別に数値の異なる、紙で提出するレポート問題を出題することも行った。課題タイプとして、1回のみ受験可能な小テスト/宿題を使用し、問題は表示するだけで解答の入力は求めない。学生は問題を紙などで解き、Möbius でない手渡しなどの方法で提出する。これは、Möbius で困難な解答過程の記述の手動評価が必要な場合や、短時間には記述が完了しない問題の場合に利用した。科目内では、連立1次方程式の解法、逆行列の計算、行列式の計算の3つについてこのような出題を行った。採点は Möbius が表示する正解を参照して教員が行い、学生へのフィードバックは小テストと同様に行った。オンライン授業に切り替えたときには、学生が自ら紙をスキャンして LMS に提出することが考えられる。

3.6 数式処理システム入門の授業

Möbius のサーバ上では数式処理システム Maple が動作して問題の生成や解答の採点を行っている。学生には、Maple の存在は隠蔽されている。しかし、学生は繰り返し解答の数式入力をするところから、意識せずに Maple の文法に習熟することになる。習熟する範囲は四則演算や平方根などの一部の関数、ベクトルや行列の表記などごく一部ではあるものの、この状況を利用し、学生に対して数式処理システムを紹介する授業を 1 講時分実施することにした。応用数学の大学教育では、数値計算と同様に、数式処理にも習熟することが必要と考えられている。

サイトライセンスにより Maple は学内の任意のコンピュータにインストール・実行可能であること、龍谷大学先端理工学部はノート PC による BYOD を導入していること、数式処理の使用は各自のデバイスで使用できてこそ活用できること、から、各学生のノート PC にインストールを行うことも考えた。しかし、Maple の実行ファイルサイズが大きき、インストールに時間と労力を要することから、この方針は採用せず、この回のみ計算機実習室で授業を行い、設置 PC にインストール済みの Maple を使用することにした。

活動としては、簡約階段行列、連立 1 次方程式の解、逆行列、行列式などを求める例のワークシートを、教員が学生に与え、実行することを求めた。さらに、同時期に出題していた、逆行列や行列式を求めるレポート問題（問題は学生別に異なる）を、ワークシート内の数値を置き換えて検算することを求めた。実行したワークシートは LMS にアップロードして提出することを求めた。

大学の線形代数の教育で数式処理システムをより野心的に活用する計画としては、Maple を最初から各学生のノート PC にインストールさせ、例えば、行列式の数学的説明と同時に行列式を計算する関数を説明して使用させる、ような方法が考えられる。このような方針に基づく線形代数の教科書もいくつか出版されている。

これを採用しなかった理由は 2 つある。ひとつは、学生は同時に初めてプログラミング言語 (C) を学んでいるところであり、2 つの言語を同時に学ぶことによる混乱を避けるためである。もうひとつは、数式処理システムに早くから触れることにより、例えば、「手計算による簡約階段化は重要でない」「理論を理解しなくても結果は得られる」などの誤解が生じるのを避けるためである。

後者については、主観的な観察では、この科目のこの講時以降にそのようなことは起きなかった。学生の多くは、複雑な手計算を注意深く行うことよりも、Maple の関数を正しく使用して計算することのほうが難しいと感じたようだった。これ以降の学期に、学生が Maple を活用する状態にはならなかった。

4 授業の評価

履修者のうち、1 年生 99 名と再履修の 11 名が単位を修得した。学期末に履修者全体に対して行ったアンケートの各選択肢への回答数を表 2 に示す。

自由記述では、「Möbius が役に立った。今後も利用していきたいと思う。」「ほぼ毎回やる小テストは Möbius の問題を解けていたら満点近い点数を取れて」というものがあつた一方、「Möbius は簡単になっていたりあくまで理解出来ているかをチェックしているだけだから」「Möbius っていうシステムがあつて、自分の理解が間違っていないか確認できるし、

表 2: 「次の要素は、小テストで高得点を取るためにどのくらい役立ちましたか?」の結果

要素	評価の 平均値	使わ な かった	1=役 立 た な か っ た	2=あ ま り 役 立 た な か っ た	3=や や 役 立 っ た	4=役 立 っ た
配布資料	3.4	0	6	2	12	35
口頭説明	3.2	1	6	4	19	25
例題の略解	3.1	1	5	5	24	20
チーム課題と略解	3.3	1	4	2	22	26
Möbius の練習問題	3.3	4	5	5	10	31
小テストの略解と解説	3.2	2	3	9	16	25
教科書	3.1	3	4	4	26	18
他の参加者との相談	3.3	2	6	5	11	31

気が済むまで練習できる」など、Möbius は誤解していないかのチェックと練習用にすぎないことを認識している解答もあった。

また、「Möbius の練習問題の解説をもう少しわかりやすくしていただけるとありがたいです。」のように、Möbius への取組のみで理解を進めることが難しいことを示唆する解答もあった。正解不正解に対する現在の Möbius の問題でのフィードバックは、主に「正解はこれです」というだけのものが多く、不正解だったときに理解に導くようなことはしていない。これは、Möbius の問題は教科書の章に対応して細かい課題に分けて提示していることから、似た例題を教科書で見つけて学習してほしいという意図である。しかし、フィードバック内に教科書の対応ページを具体的に示したり、過程付き解答そのものを載せたりすることで、Möbius を起点に、学生の理解が進む可能性はある。

各学生の Möbius の問題が正解に至った比率 (Möbius 比率) と小テスト合計点数 (最大 90) の散布図を示す (図 5)。相関係数は 0.434 だった。Möbius 比率の学生平均は 38.8%、標準偏差が 26.0% だった。

比率が 1 でありながら点数が 40 点未満の学生が 1 名いるが、Möbius の取り組みに応じて小テストは一定以上の得点が得られる、Möbius なしでも小テストで得点できる学生もいる、ことを示唆する。これが事実であれば、一部の小テスト低得点の学生に、Möbius の利用をより強く薦めれば、小テストの得点が上昇する可能性がある。

5 まとめ

任意の時点で対面とオンラインが切り替え可能な形の線形代数の授業のデザインと実践について述べた。どちらの授業形態であっても、オンライン数式入力演習・評価システム Möbius はいくつかの活動において有効に機能することが示唆される。今後、社会の状況によっては、急な切替は考慮の必要がなくなるかもしれない。そうであっても、この実践の記録が、対面とオンラインを複合した形態の授業デザインの検討に役立つことを期待する。

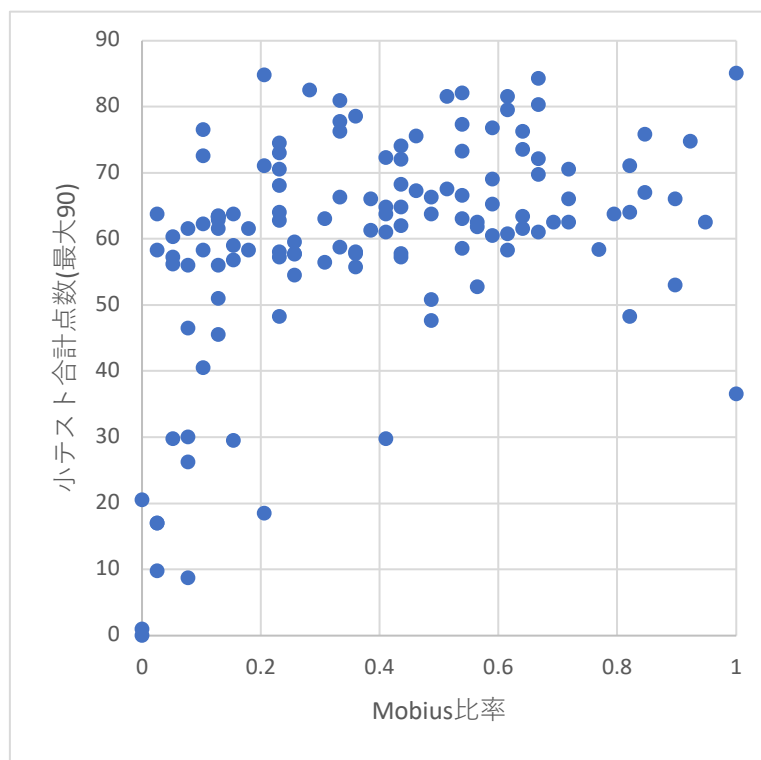


図 5: 各学生の Möbius 比率と小テスト合計点数の散布図

Möbius に関して，ここでは，解答のみを数式で入力する，即時に自動採点されるという特徴をとらえて考察した．さらに詳細には，この枠内でどのような問題を出題するか，どのようにランダム化するか，どのようなフィードバックをするかという点を検討する必要がある．

謝辞

本研究の一部は科研費（21H04412, 21H00921）の助成を受けた．

参考文献

- [1] DigitalEd, Möbius <https://www.digitaled.com/mobius/>. (閲覧日: 2023 年 6 月 30 日)
- [2] 加藤文元, 大学教養 線形代数, 数研出版 (2019).
- [3] 今本恕, 高田秀志, 紙媒体を用いた作業の特長を誘発する協調検索支援システム, 情報処理学会論文誌 **59** (2018) 723–730.
- [4] 中村泰之, ペタ語義: 数学 e ラーニングシステムの国内外の動向と今後の展望, 情報処理 **64** (2023) 118–121.

- [5] Maplesoft, User Case Study: University of Waterloo Improves Learning while Saving Money with Maple T.A., <https://www.maplesoft.com/company/casestudies/stories/102384.aspx> (閲覧日: 2023年11月22日)
- [6] André Heck, Leendert van Gastel, Diagnostic Testing with Maple TA, WebALT 2006 Proceedings (2006) 37–51.
- [7] Gordon Clement, Technology in Mathematics Education: Implementation and Assessment, Master thesis, University of Guelph (2011).
- [8] David Smith, Aron Pasieka, Ralf Becker, Christina Perdikoulis, Student Success in Asynchronous STEM Education: Measuring and Identifying Contributors to Learner Outcomes, in 2022 IEEE Global Engineering Education Conference (2022) 473–479.
- [9] 北本卓也, Maple T.A. の授業援用について, 数理解析研究所講究録 **1907** (2014) 182–187.
- [10] 和久井道久, Möbius のためのコースモジュール—授業関連, <https://www2.itc.kansai-u.ac.jp/~wakui/presen721.html> (2020). (閲覧日: 2023年6月30日)
- [11] Chris J. Sangwin, Computer Aided Assessment of Mathematics, (2013) Oxford University Press.
- [12] 樋口三郎, 数式入力による数学評価システム Maple T.A. を利用した理工系学部での基礎教育, 数理解析研究所講究録 **1978** (2015) 72–78.
- [13] 樋口三郎, 数学オンラインテスト Maple T.A. を用いた大学入学前数学教育の実践, 教育システム情報学会 第41回全国大会論文集 (2016) 186–187.

A Möbius の問題例

問題「行列 A の簡約階段化と階数を求めなさい」の Algorithm 欄に記載する, 3.4 に沿って作成したソースコードの例をリスト 1 に示す. 簡約階段化の正解は ans1 , 階数の正解 ans2 , 採点用関数はそれぞれ grade1 , grade2 である.

リスト 1: 簡約階段化と階数を求める問題

```

1 : size of a matrix ;
2 $n=range(3,5,1);
3 $m=range(3,5,1);
4
5 : random answer ;
6 $A0=maple("randomize(): LinearAlgebra[RandomMatrix]($m,$n,density=0.4,
    generator=-2..2)");
7 $ans1=maple("LinearAlgebra[ReducedRowEchelonForm]($A0)");
8 $ans2=maple("LinearAlgebra[Rank]($A0)");
9
10 : elementary operations ;
11 :: row indices ;
12 $i1=range(1,$m,1);
13 $j1=range(1,$m,1);
14 condition:ne($i1,$j1);
15 $i2=range(1,$m,1);
16 $j2=range(1,$m,1);
17 condition:ne($i2,$j2);
18 condition:ne($i1,$i2);
19 $i3=range(1,$m,1);
20 $j3=range(1,$m,1);
21 condition:ne($i3,$j3);
22 $i4=range(2,$m,1);
23 :: coefficients ;
24 $a1=range(2,5,1);
25 $a3=range(2,5,1);
26 $c=range(2,5,1);
27 :: application ;
28 $A1=maple("Student[LinearAlgebra][AddRow]($ans1,$i1,$j1,$a1)");
29 $A2=maple("Student[LinearAlgebra][AddRow]($A1,$i2,$j2,$a3)");
30 $A3=maple("Student[LinearAlgebra][SwapRow]($A2,$i3,$j3)");
31 $A4=maple("Student[LinearAlgebra][MultiplyRow]($A3,$i4,$c)");
32
33 : conversion to MathML ;
34 $disp=maple("proc(A) printf(MathML[ExportPresentation](A)); end proc;")
    ;
35 $A4disp=maple("$disp($A4)");
36 $ans1disp=maple("$disp($ans1)");
37 $ans2disp=maple("$disp($ans2)");
38
39 : grading codes ;
40 $grade1=maple("proc(A,R) LinearAlgebra[Equal](A,R); end proc;");
41 $grade2=maple("proc(A,R) evalb(A-R=0) end proc;");
42
43 : test ;
44 $test11=maple("($grade1)($ans1,$ans1)");
45 $test12=maple("not ($grade1)($ans1,$ans1+$A4)");
46 $test21=maple("($grade2)($ans2,$ans2)");
47 $test22=maple("not ($grade2)($ans2,-3)");

```
