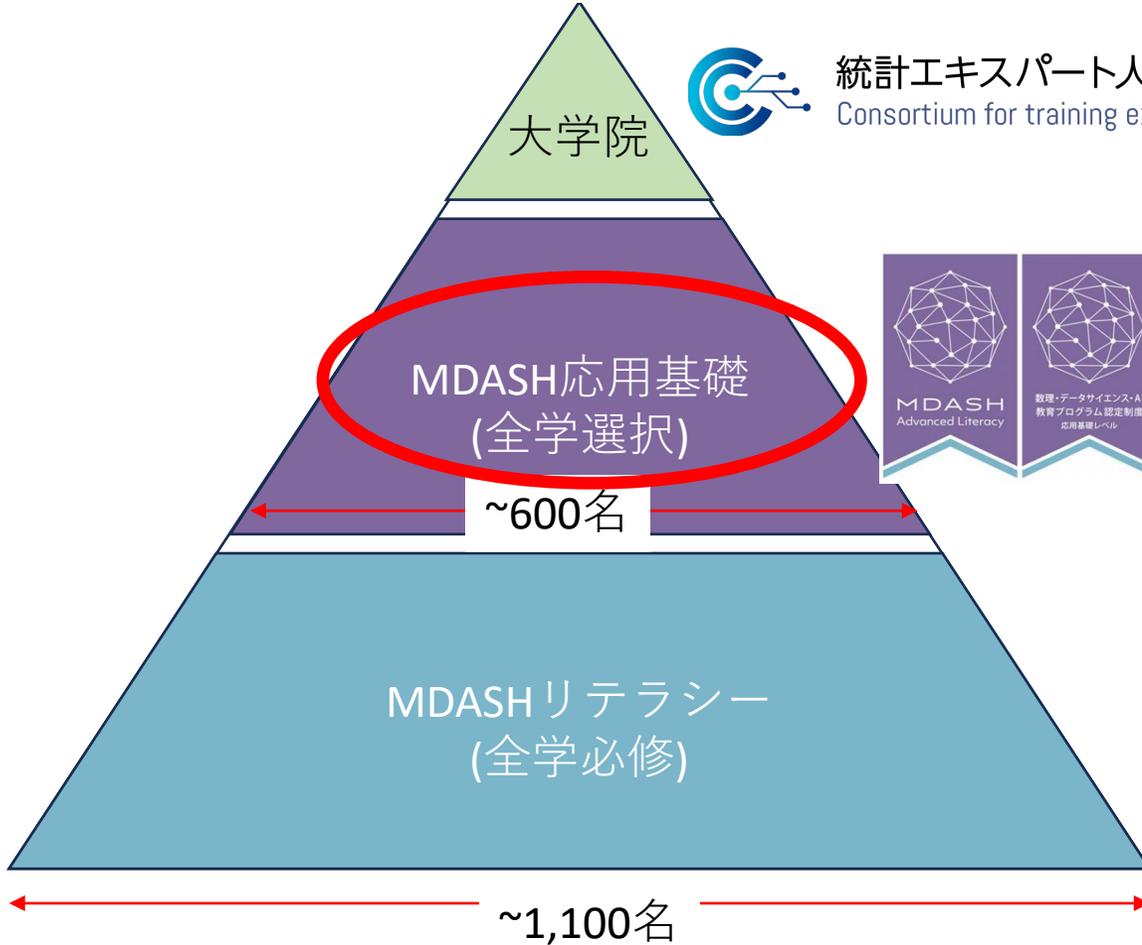


CoursewareHubを用いた大人数 プログラミング教育

群馬大学
数理データ科学教育研究センター
青木 悠樹



統計エキスパート人材育成コンソーシアム
Consortium for training experts in statistical sciences



MDASH
Advanced Literacy
Approved Program for Mathematics,
Data science and AI Smart Higher Education,
designated by the Gov of Japan
数理・データサイエンス・AI
教育プログラム認定制度
応用基礎レベル
群馬大学 (GUNMA UNIVERSITY)
認定の有効期限 : 2028年3月31日

- Python入門
- DS応用
- DS・ML・AI



MDASH
Literacy
Approved Program for Mathematics,
Data science and AI Smart Higher Education
数理・データサイエンス・AI
教育プログラム認定制度
リテラシーレベル
認定の有効期限 : 令和8年3月31日

MDASH+オープンバッジ



200-300名規模の文理横断学習者を
教員1名で正しく成績評価

Python入門について



- 全学選択科目(学部1年以上, 受講者236名)
- 教員 1 名, TA無し
- 講義型授業は学力のばらつきのため不可
- オンデマンド授業ではモチベーションの維持が困難



学習者の実行ログを記録する特化型LMS(CoursewareHub)を使用した
対面オンデマンド型

CoursewareHubとは

国立情報学研究所が開発した講義に特化したJupyterHubのディストリビューション

Jupyter(電子ノート)におけるcell(記入欄)ごとの**実行ログが記録**される仕組み



学内設置サーバー



クラウドでもオンプレでも使える
LTI認証連携ができる

大人数プログラミング教育の難しさ

Google Colaboratoryで実施(2022年度)

教員側：学生の進捗度がリアルタイムで把握できない



学生側：教員が把握できていないことに気づく



学生側：課題さえ出せばいいことに気づく
教員に質問しない



学生側：コピーしたコードを提出してもバレないだろうと思う



教員側：コピーした課題が大量に提出され、疑心暗鬼になる

CoursewareHubで実施(2023年度)



リアルタイムで進捗を把握している



教員に見られている！という意識



実行ログで判断するため、コピーは採点されない



教員側の熱意が伝わった一方で
授業中の質問件数が圧倒的に増える
(~10→~200件)

テキスト

1 前回の演習問題の解説

```
入力 [1]: import IPython.display
          IPython.display.YouTubeVideo('QieKx6vGY7Y')
```



解説動画, ~5本

2 はじめに

以下のコードは10秒に一回、自動保存するコードです。
はじめに実行してください。

```
入力 [ ]: %autosave 10
```

3 タプル

[こちらの動画](#)を見ながら進めてください。

タプルの書式は以下です。

```
(要素1, 要素2, 要素3) #tuple
```

リストの書式とよく似ていますね。

```
[要素1, 要素2, 要素3] #list
```

タプルとリストの違いはなんですか？以下のタプルとリストを用意して調べてみましょう。

```
tuple1 = ('トマト', 'いちご', 'キャベツ')
list1 = ['トマト', 'いちご', 'キャベツ']
```

```
入力 [ ]:
```



動画の補足資料



練習, ~10個

+ 宿題

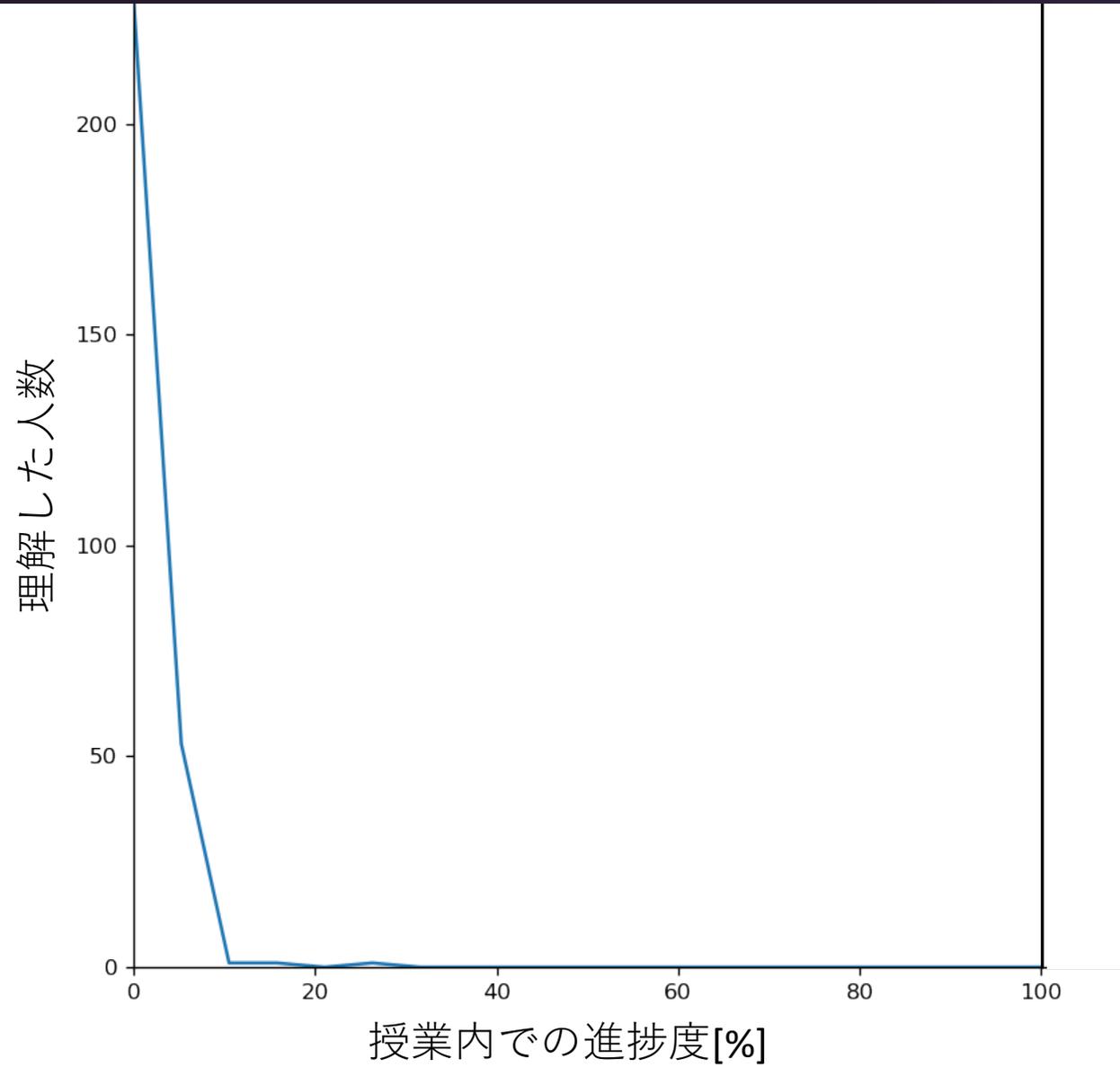
3.0.1 調査1: 型の違い

```
入力 [ ]:
```

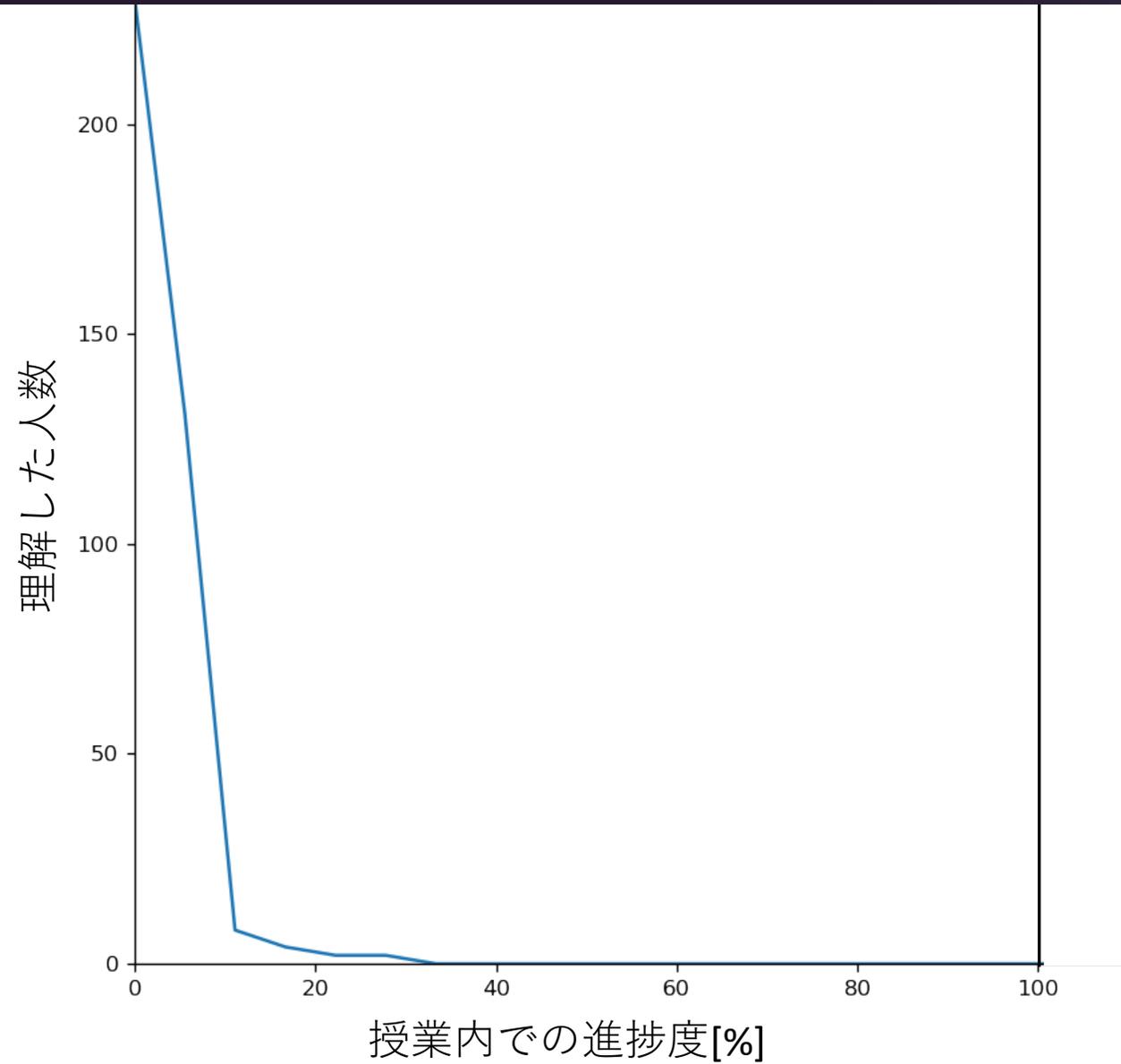
授業中の学習者の進捗度をモニター

練習問題の実行結果が”OK”であれば理解したとみなす

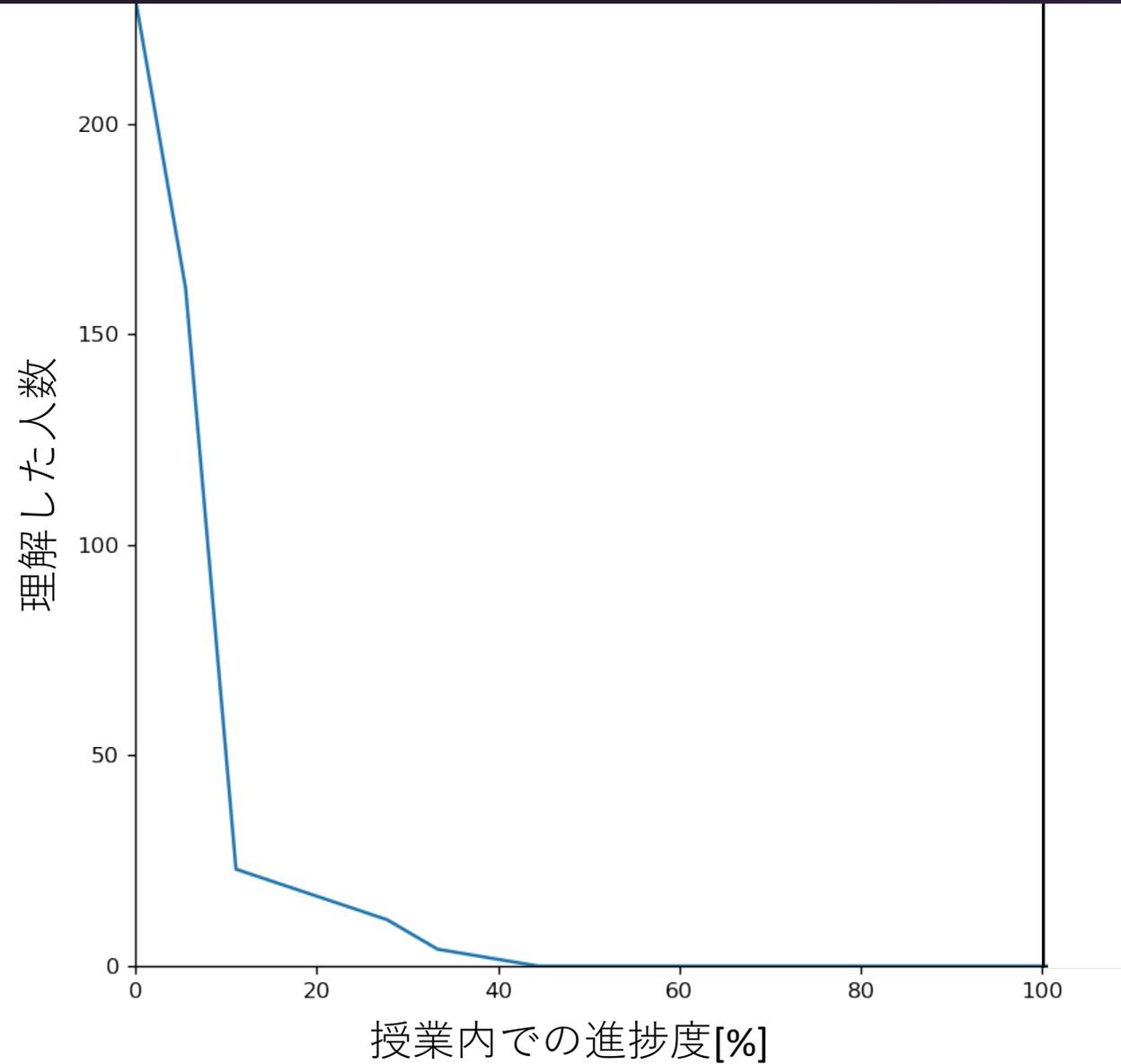
0分



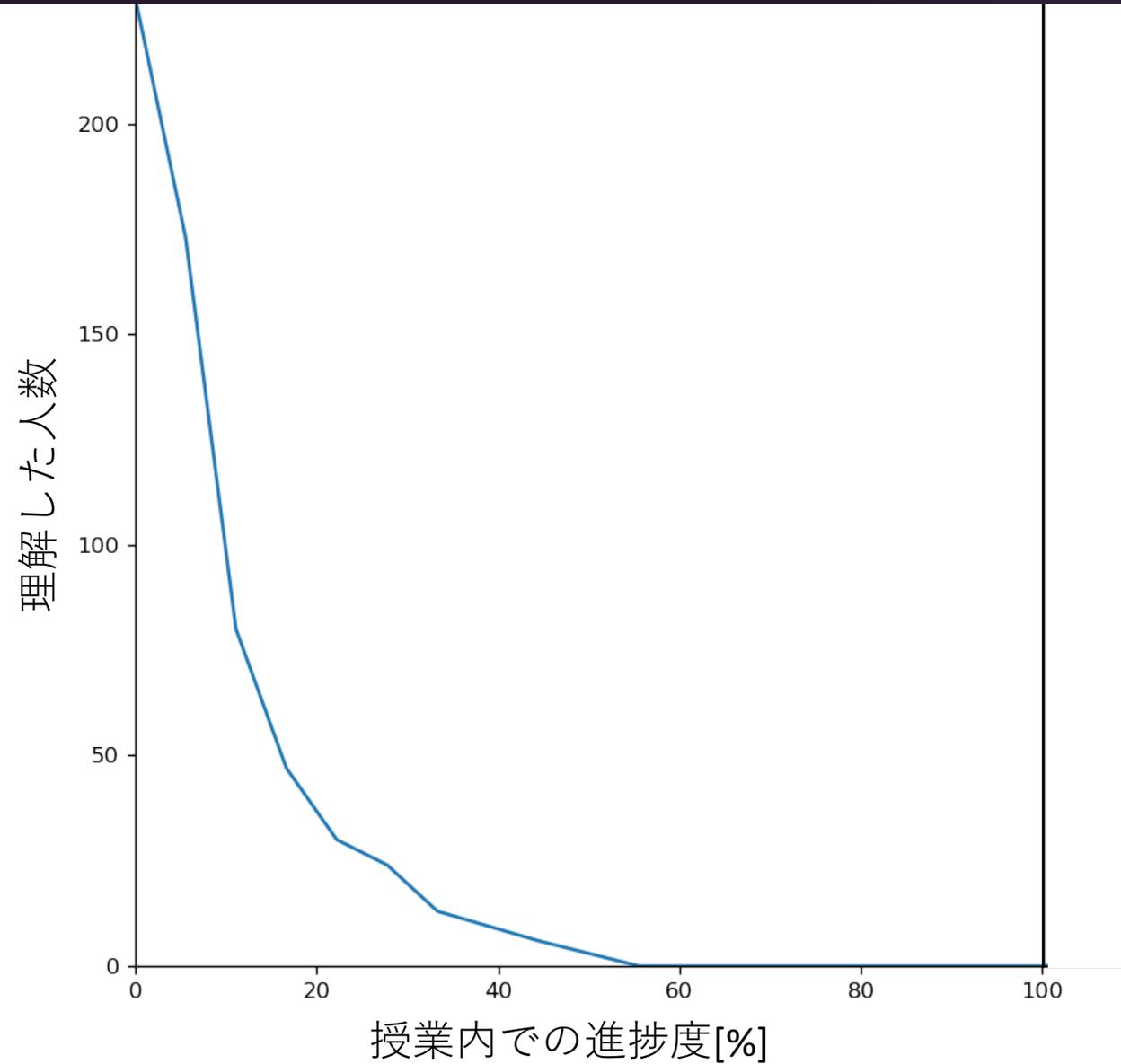
10分



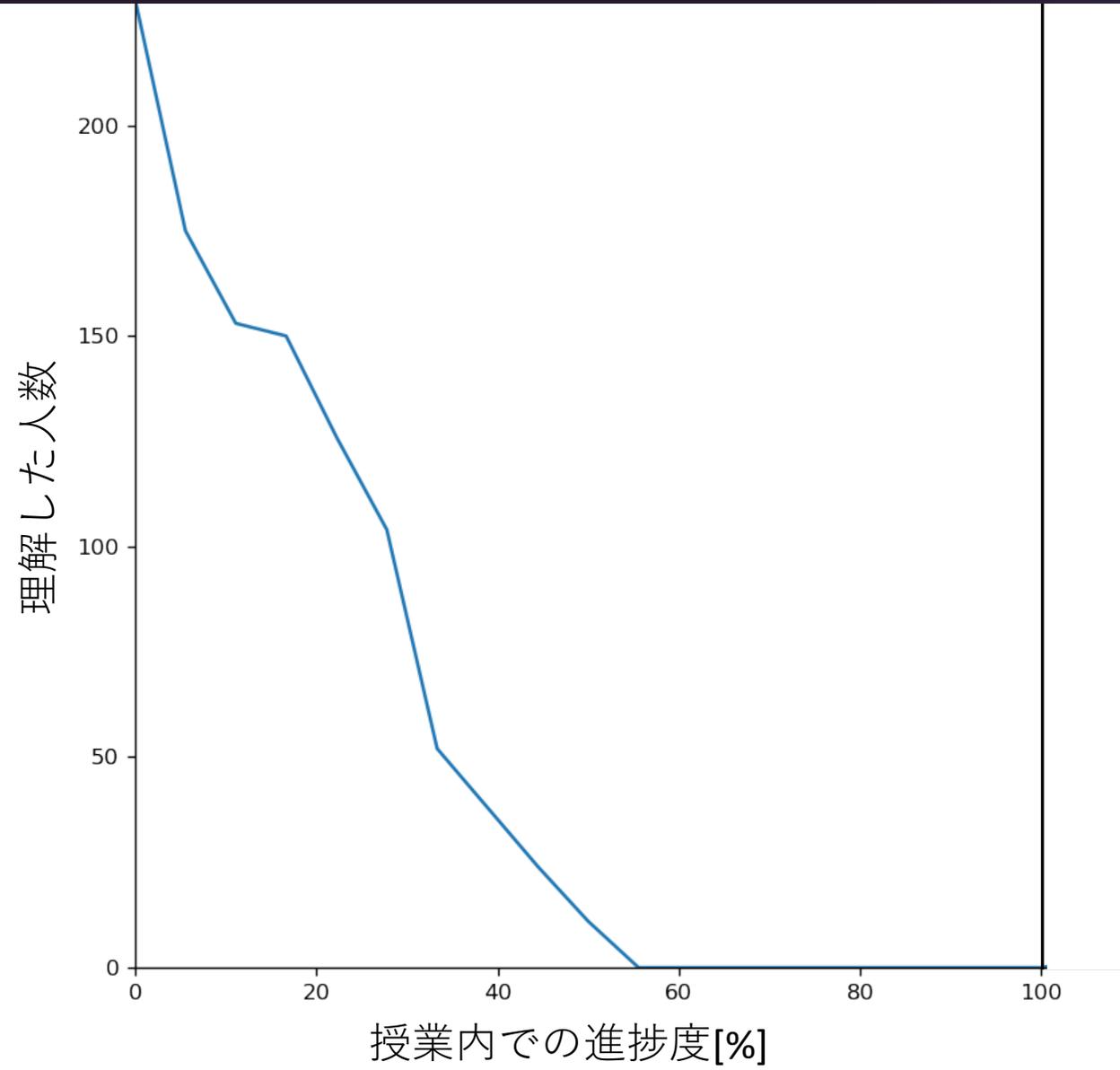
20分



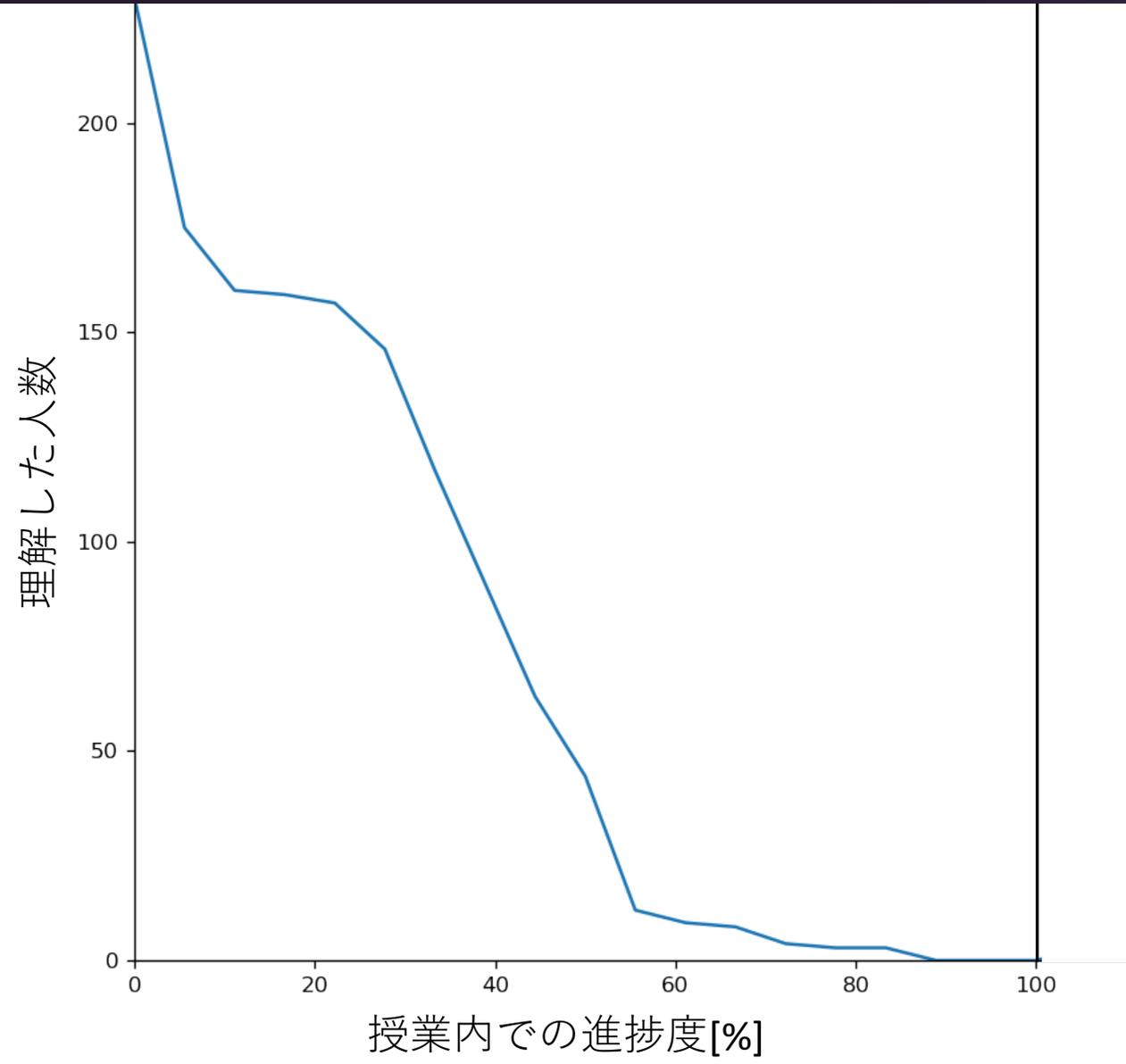
30分



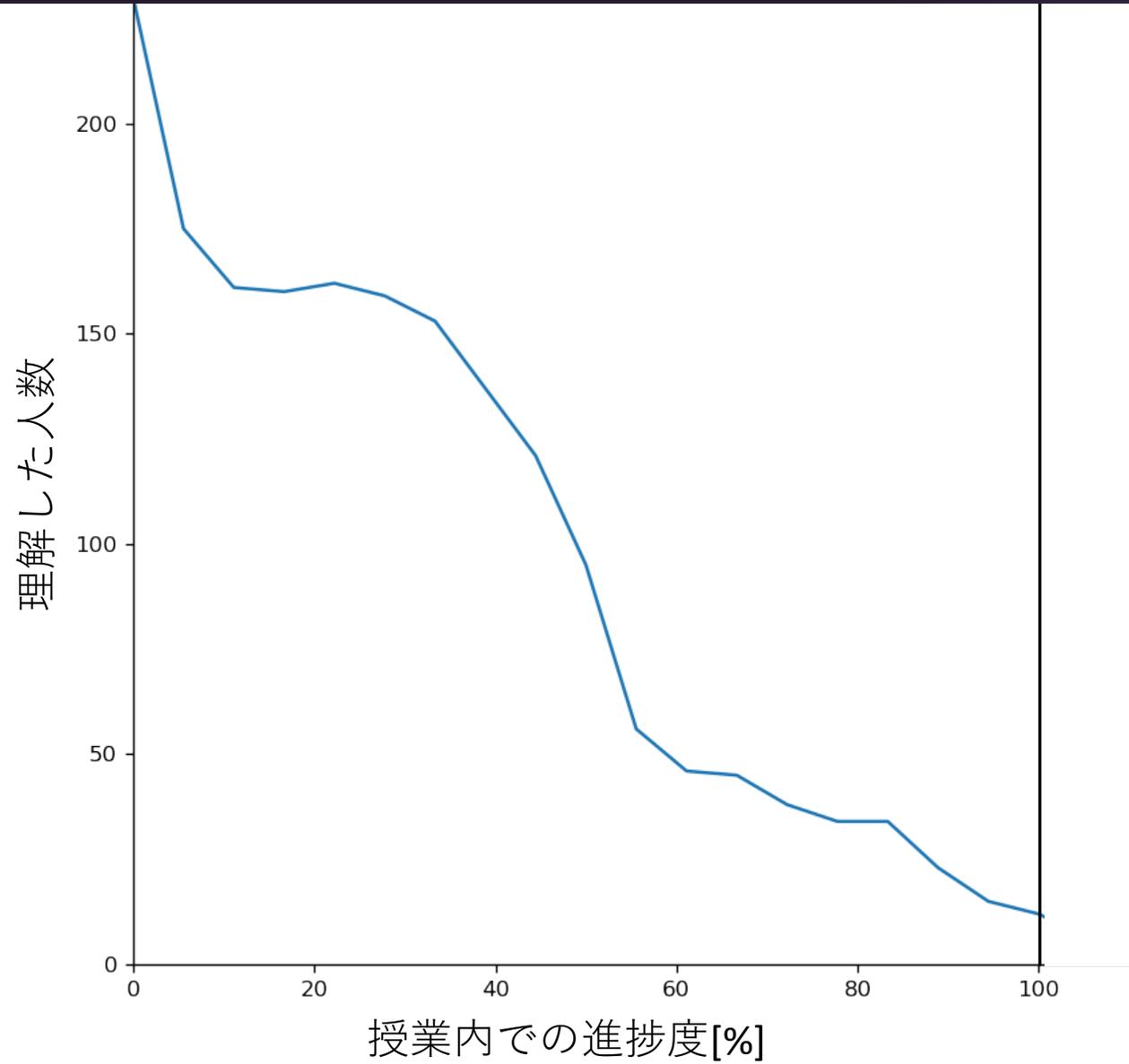
40分



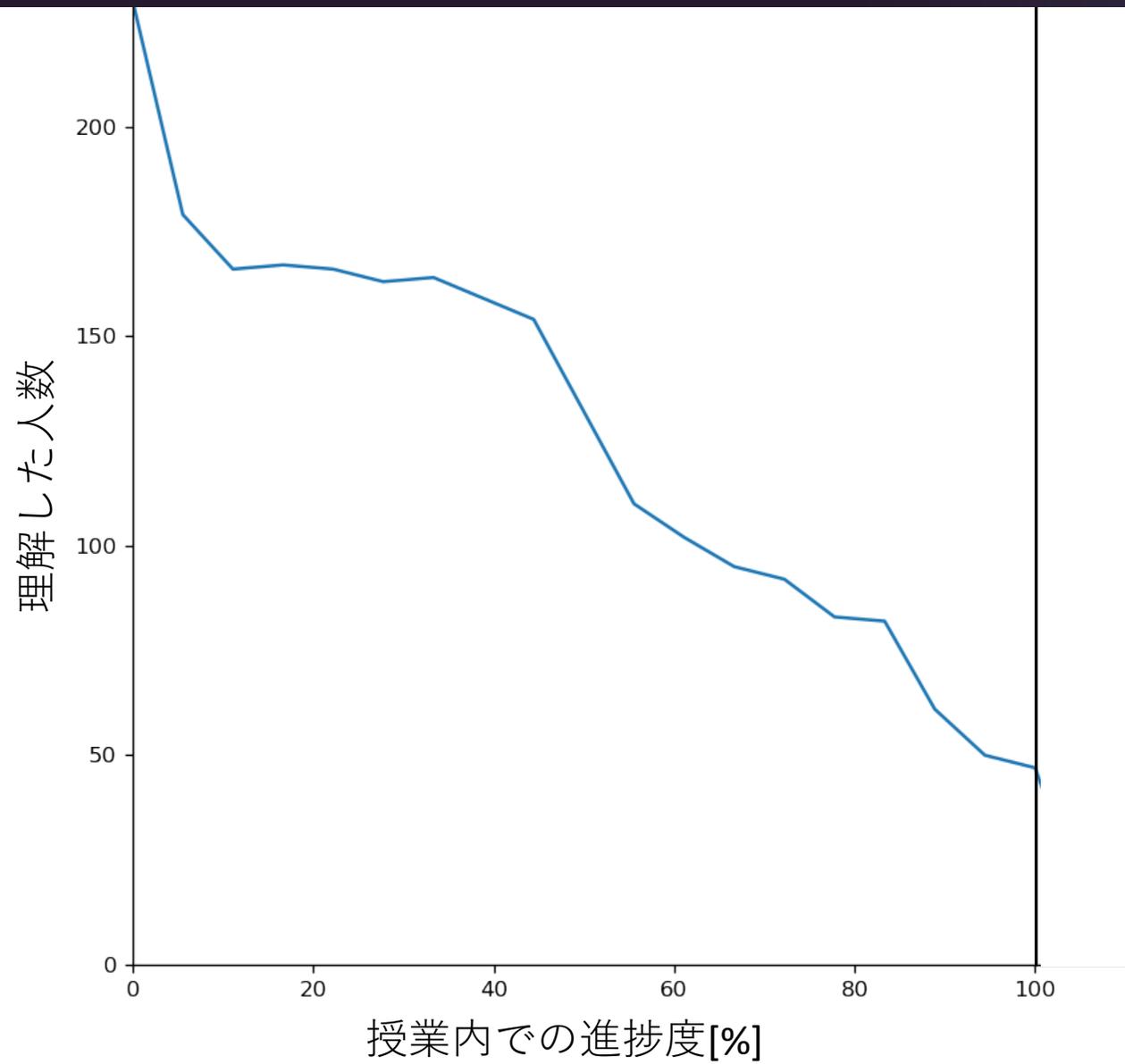
50分



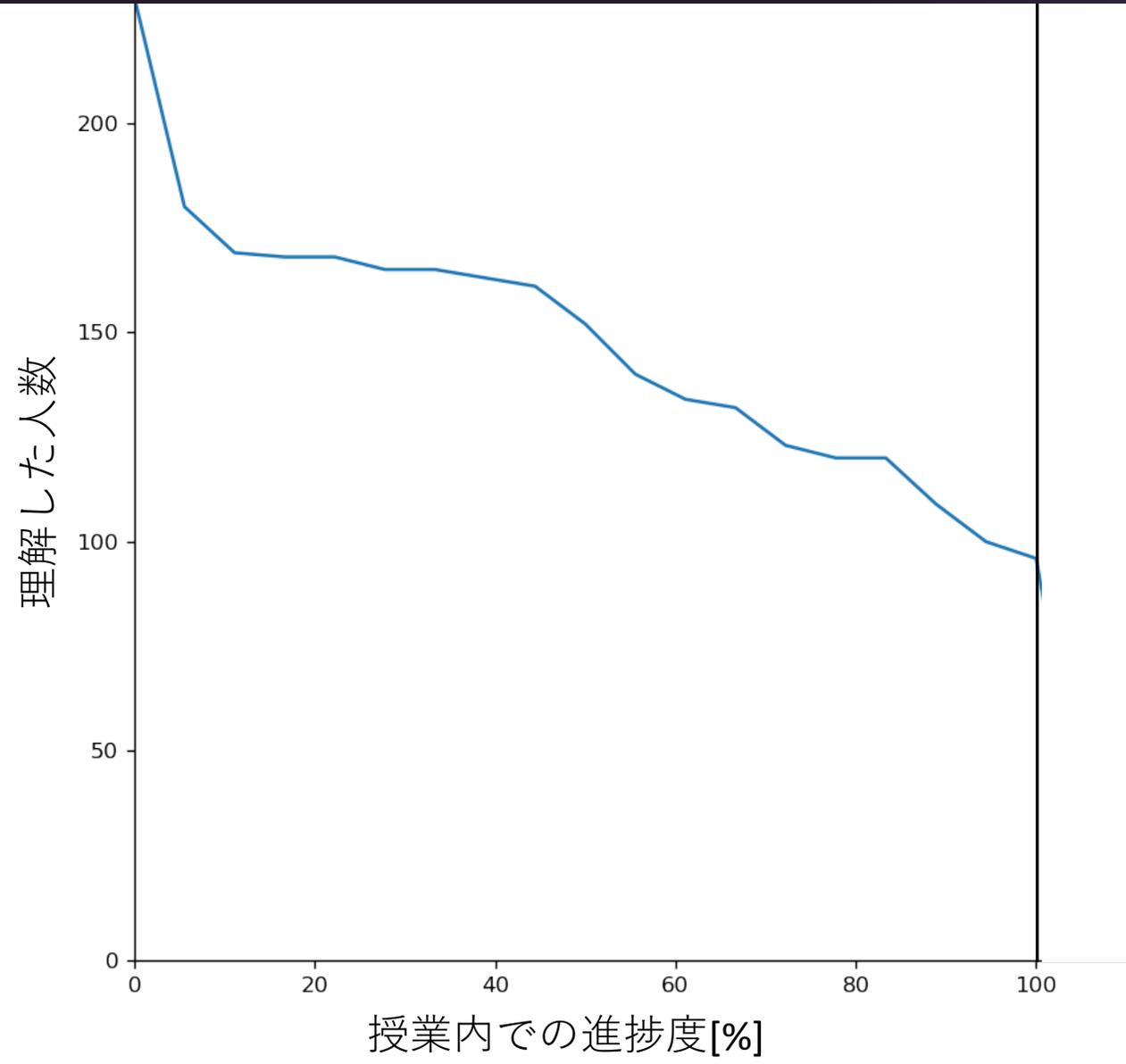
60分



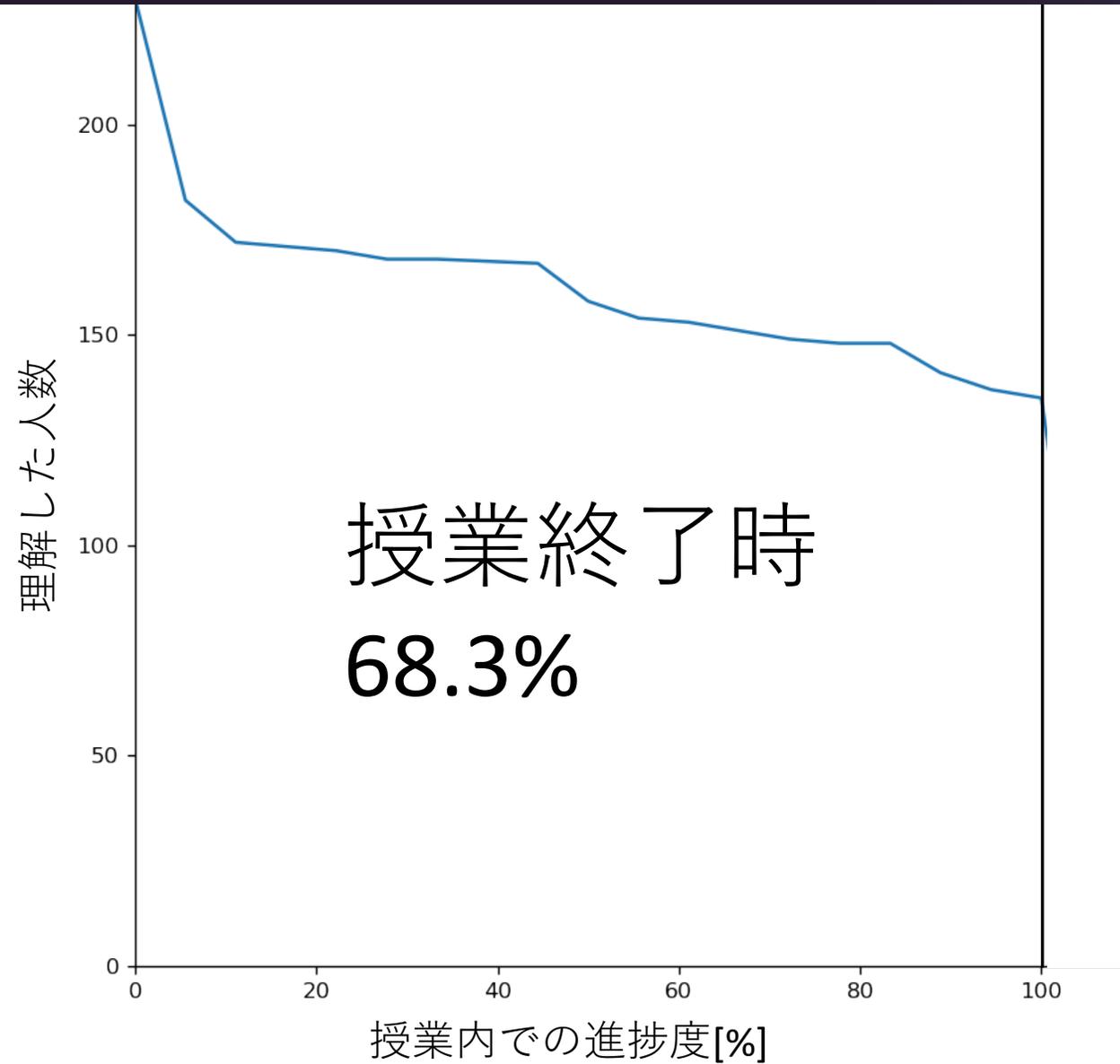
70分



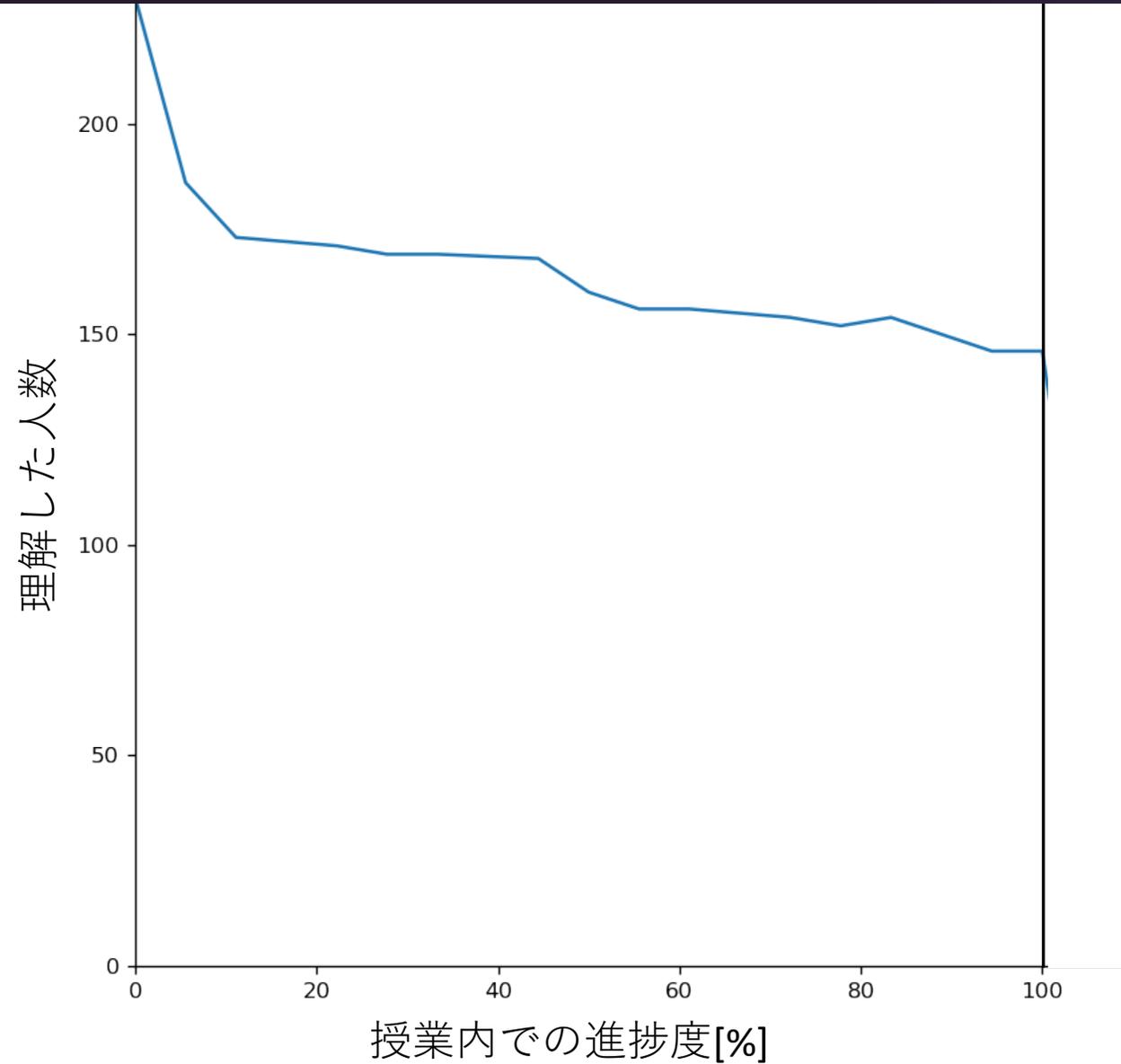
80分



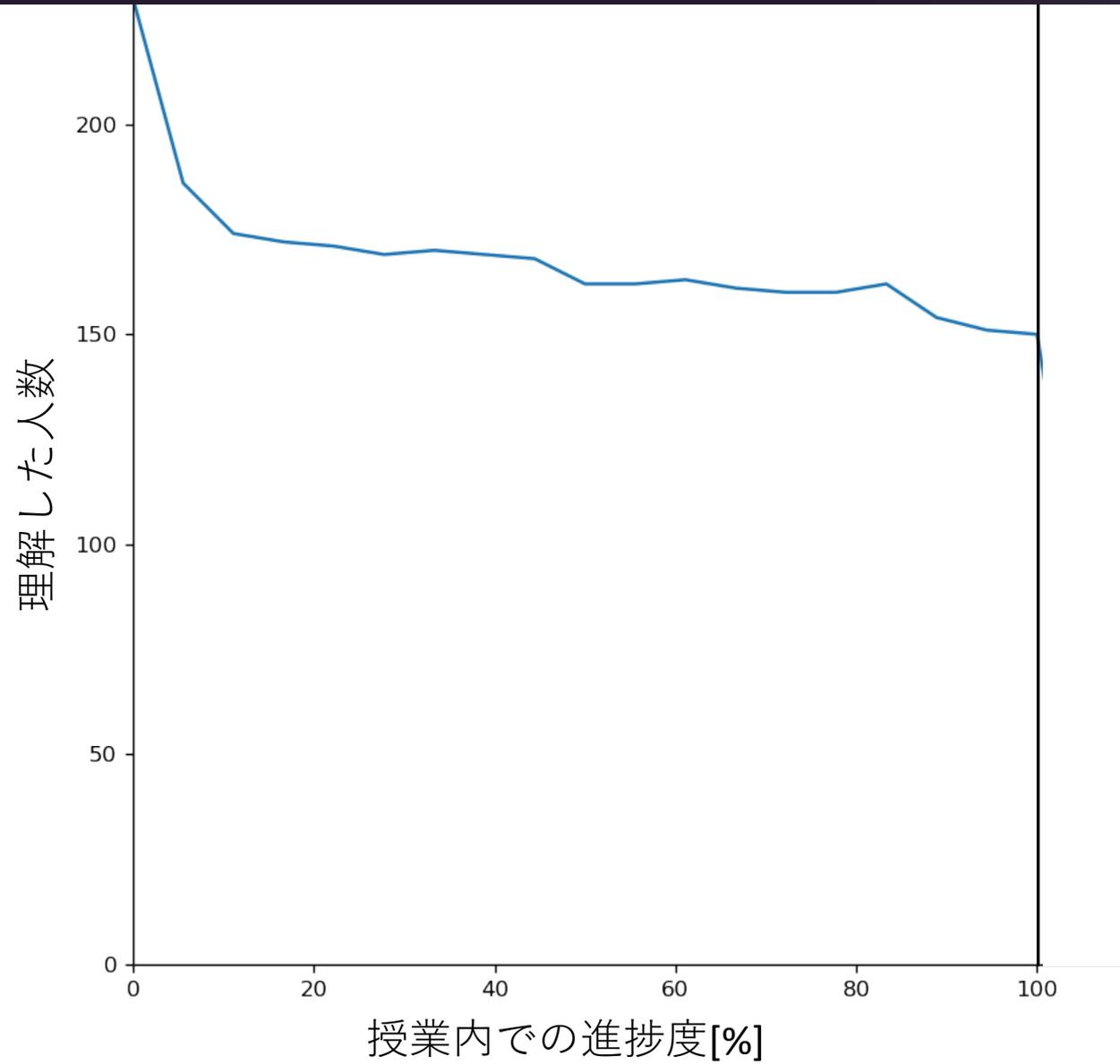
90分



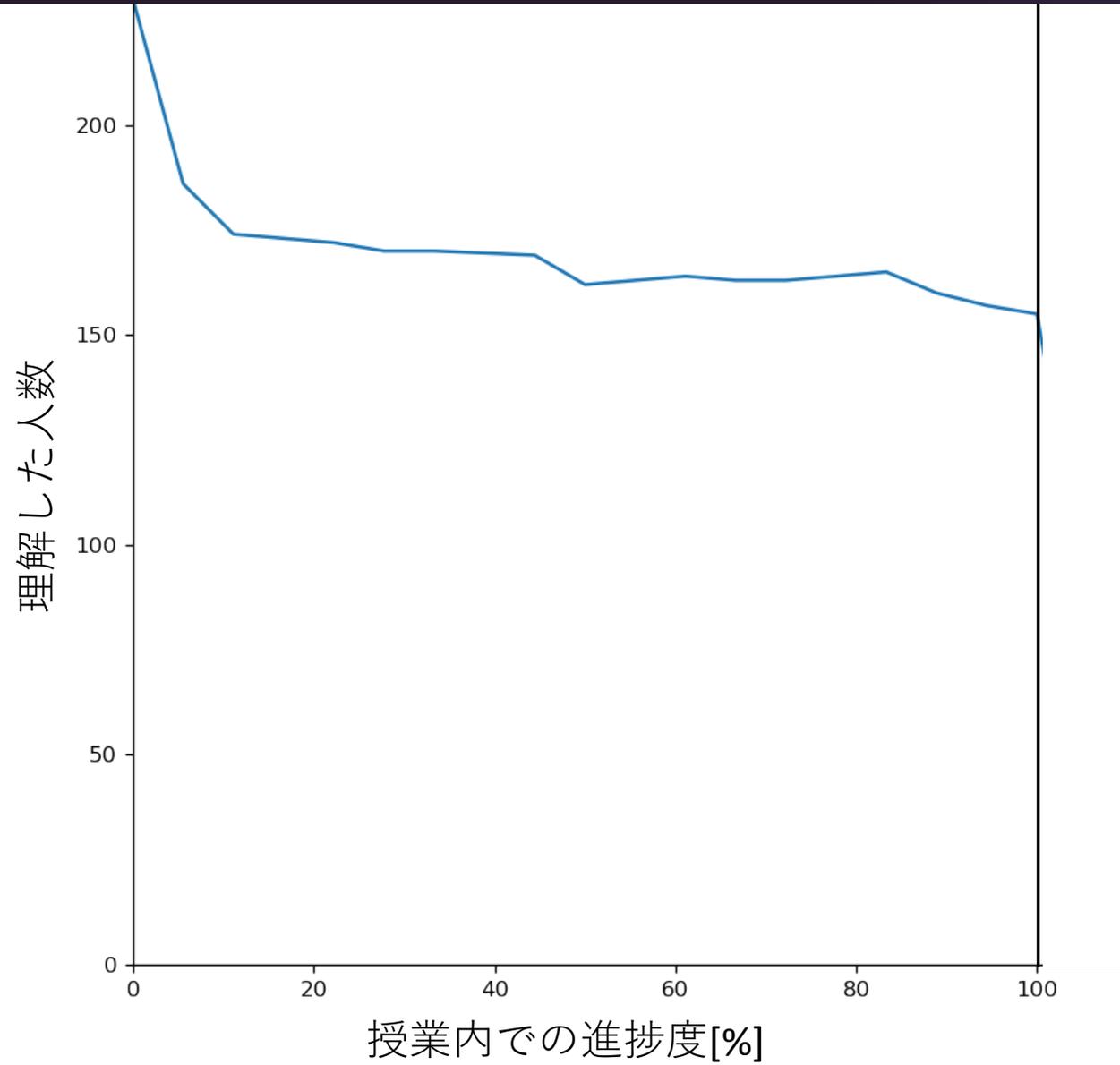
100分



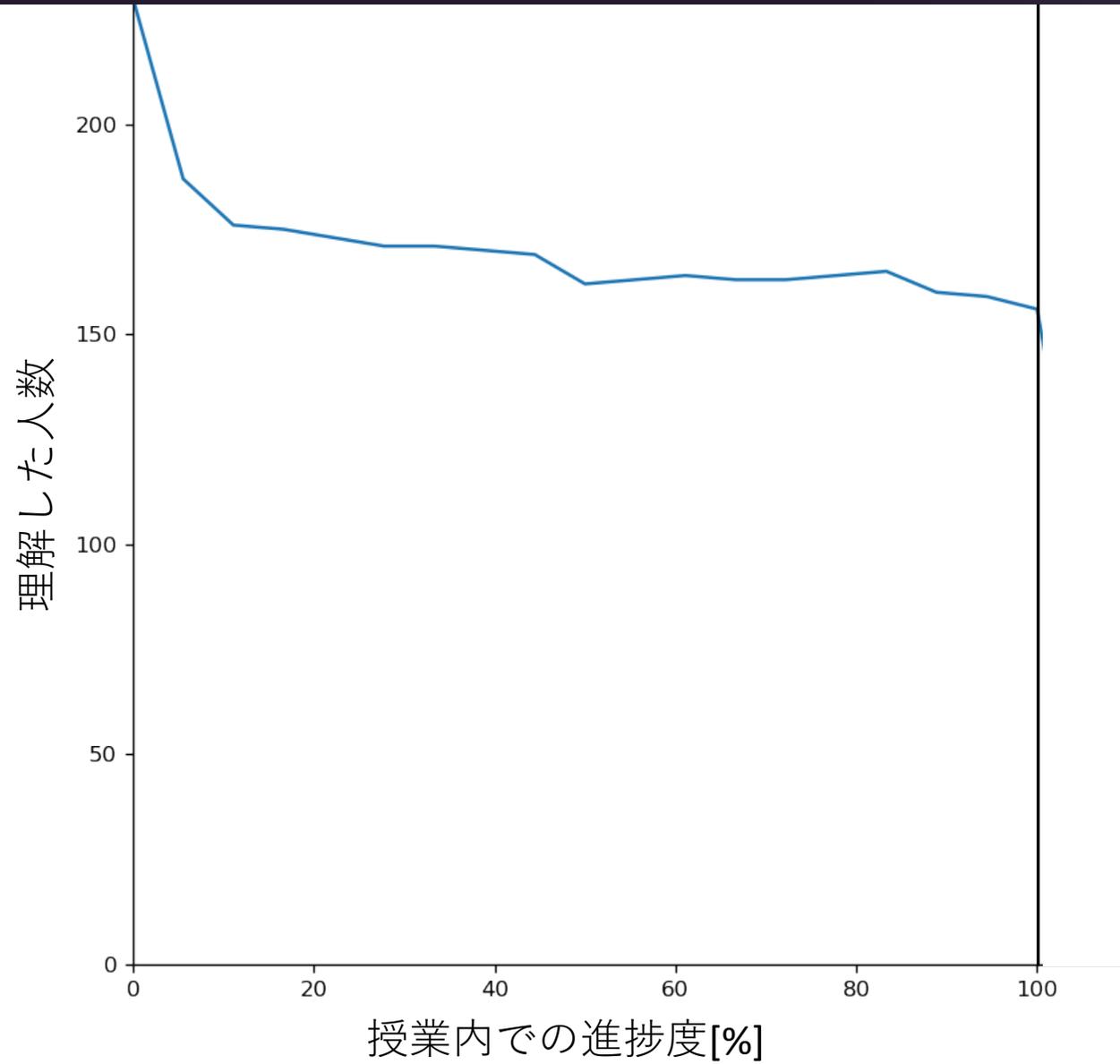
110分



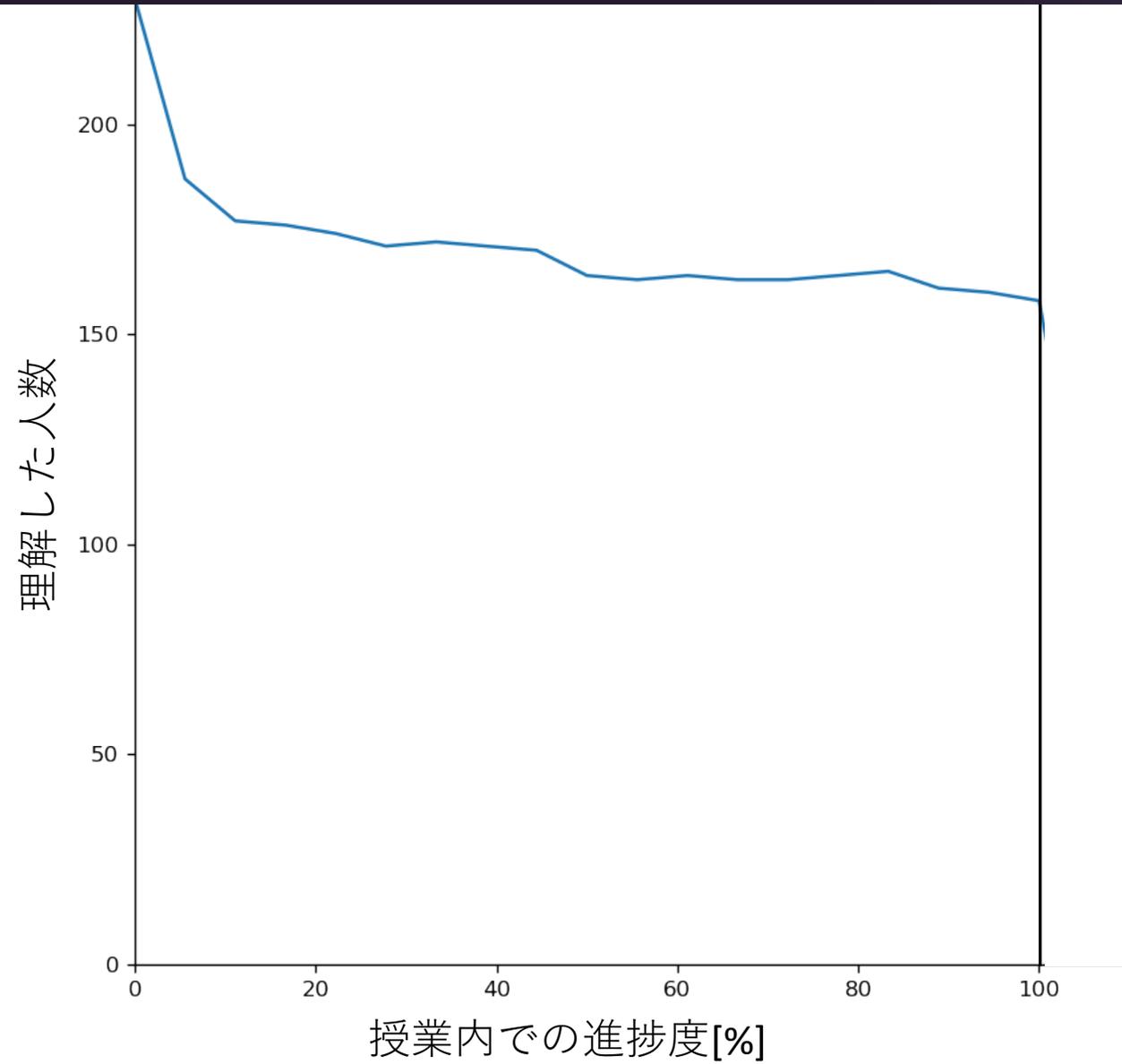
120分



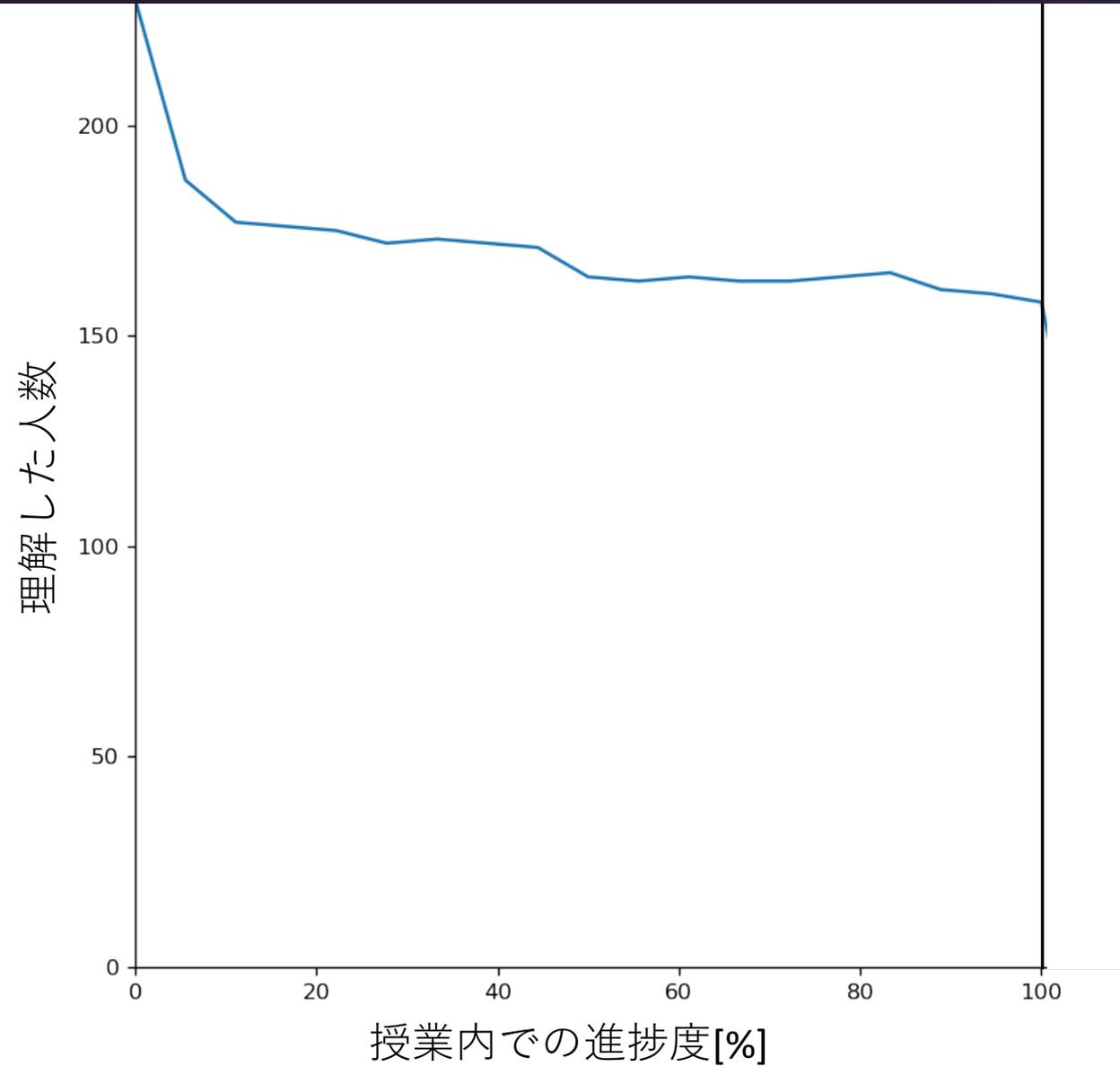
130分



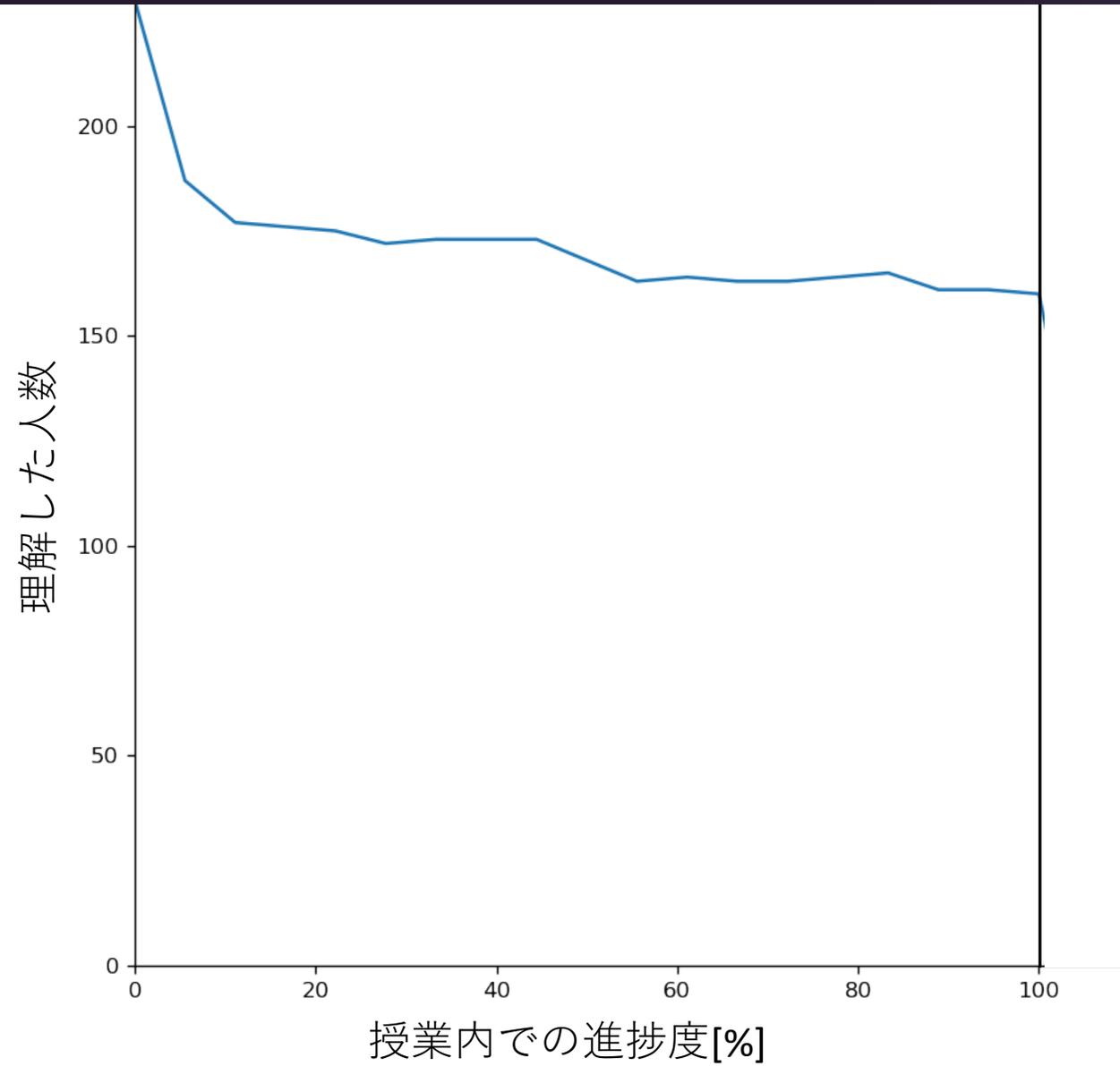
140分



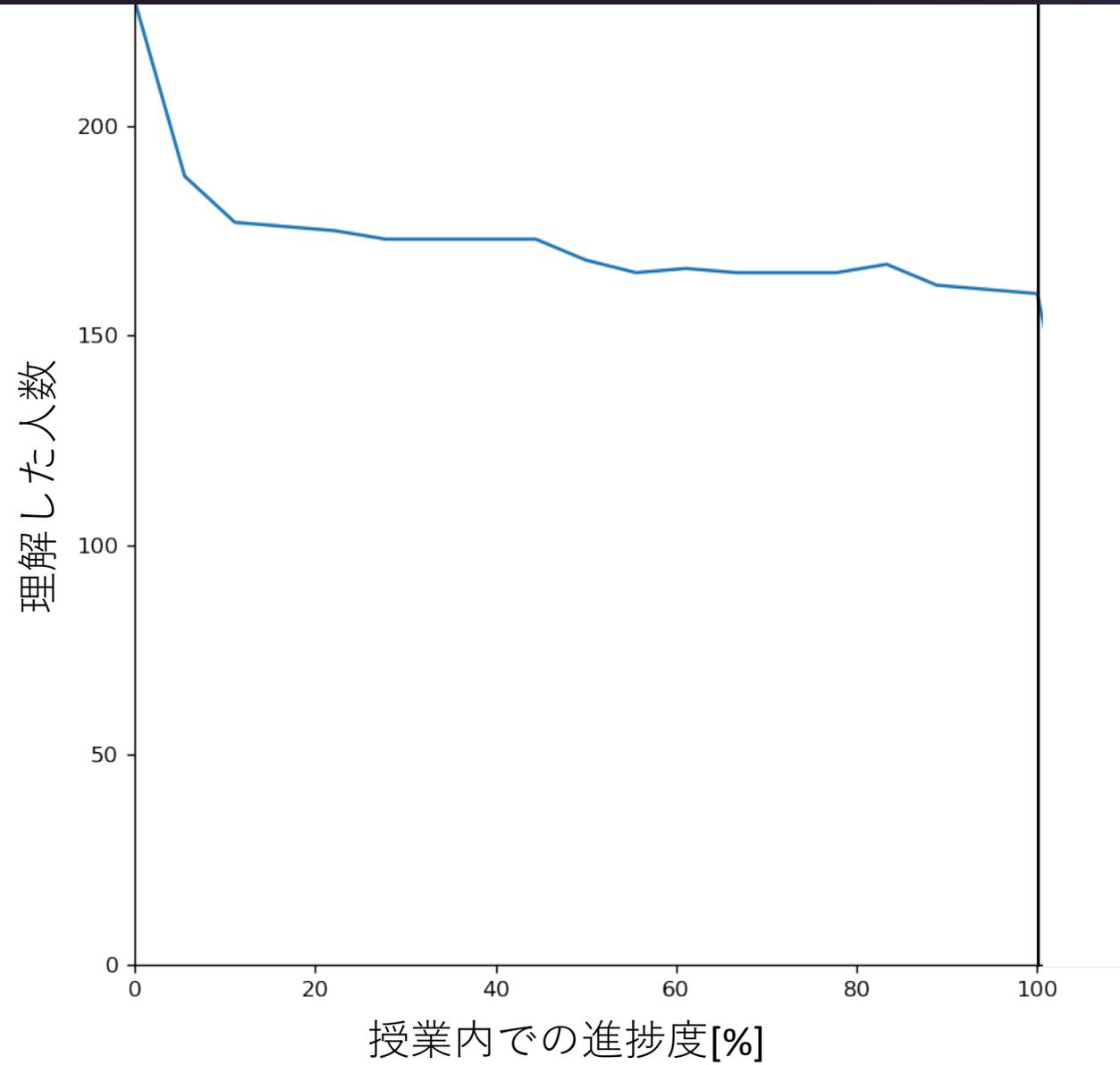
150分



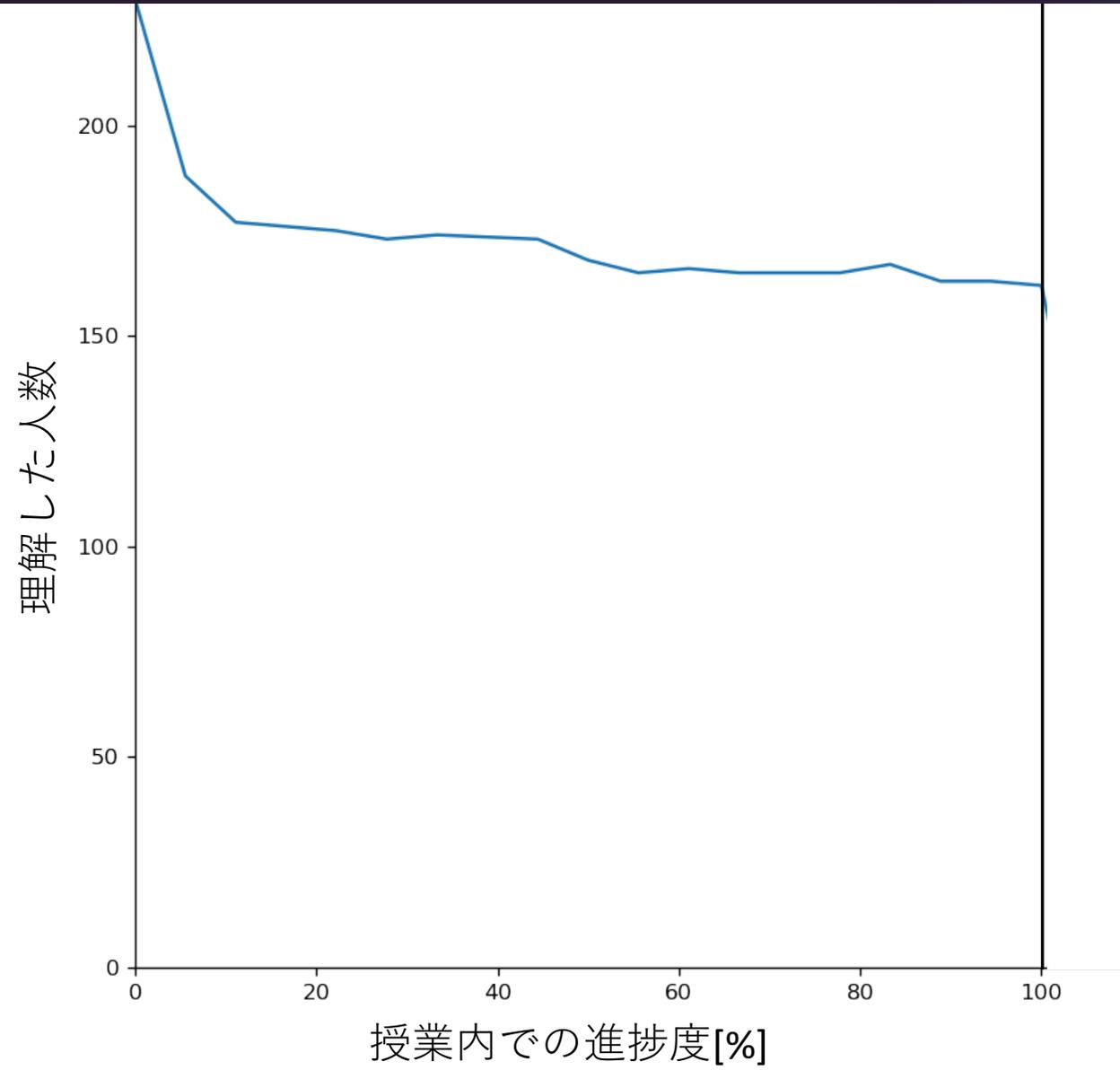
160分



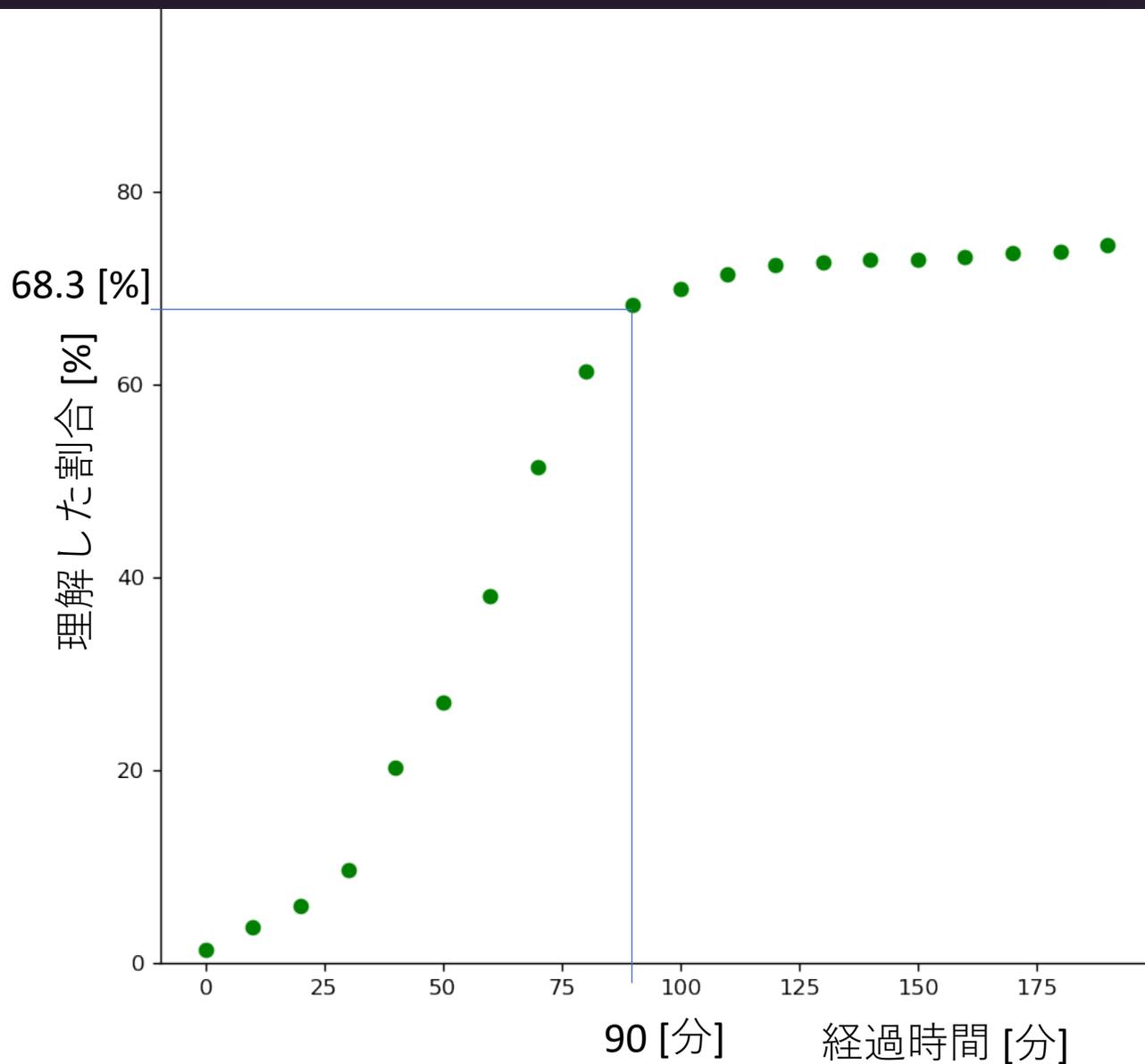
170分



180分



「練習」の時間変化



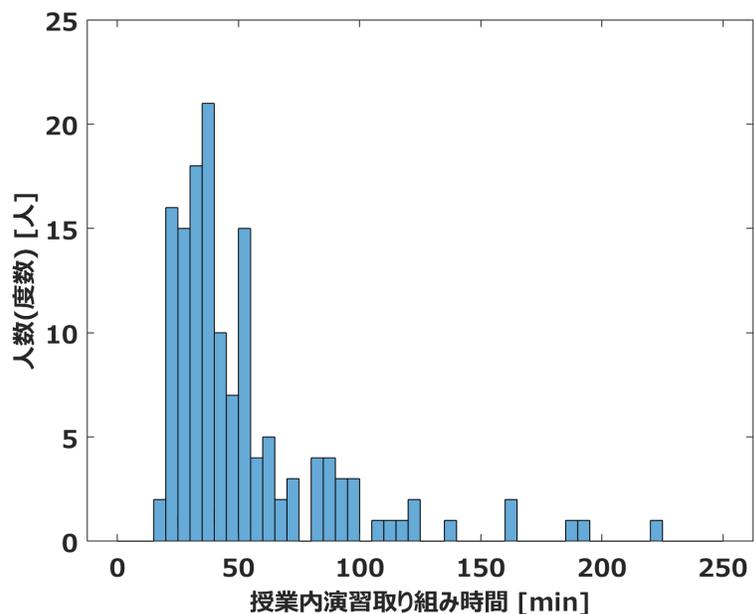
授業時間内にテキスト
の68.3%を受講完了



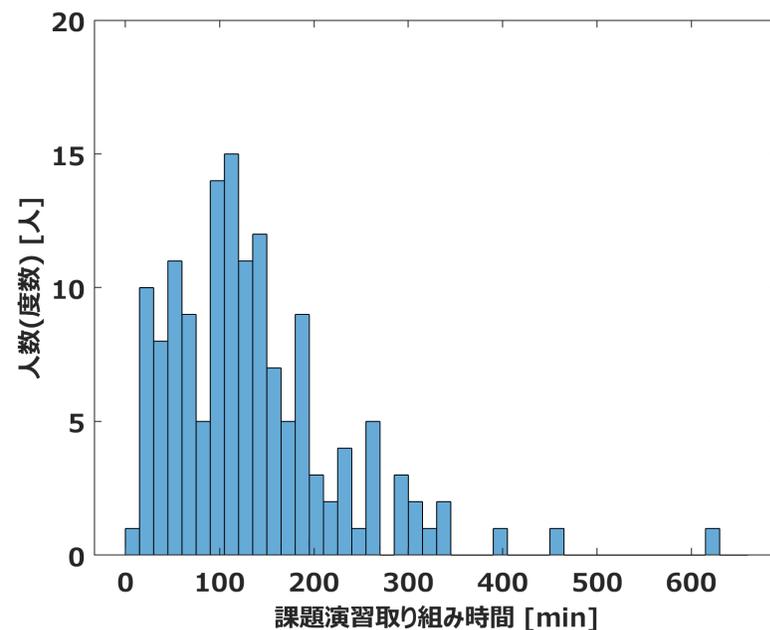
ノート提出時は90.0%

宿題のログデータの持つ情報

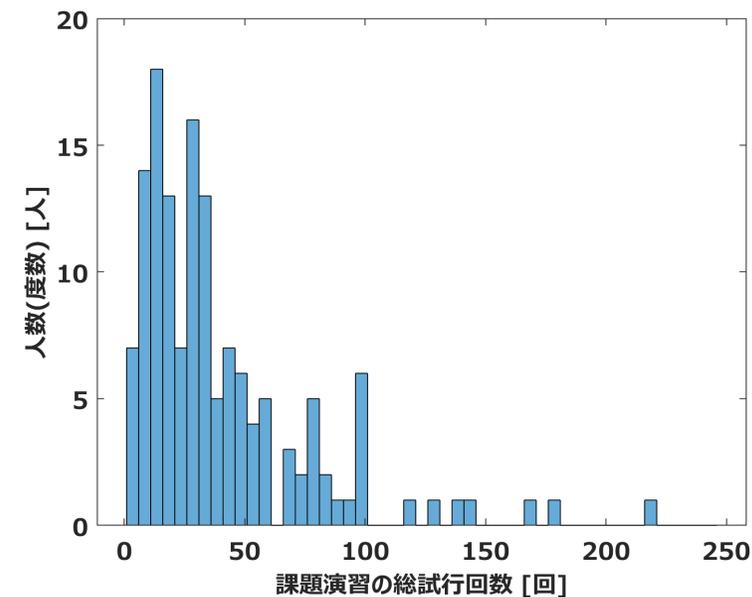
- ◇学生がいつ, どのようなコードを実行したかが分かる
 → ファイルの生成時間や内容から「取り組み時間」と「トライ・エラー回数」が得られる



授業内の取り組み時間



授業外の取り組み時間

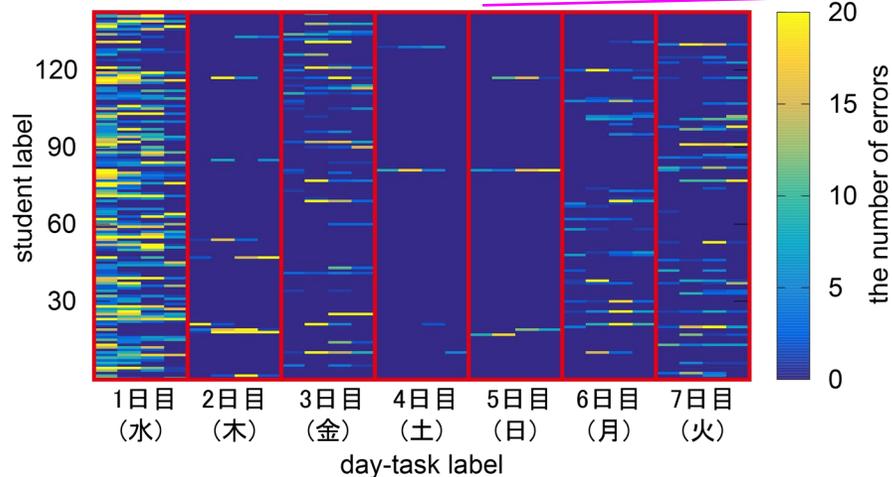


演習課題の試行回数

試行回数から何が見える？

◇学生の試行錯誤のパターンを抽出したい

アクセスが学内限定
→ 休日の取り組みは少



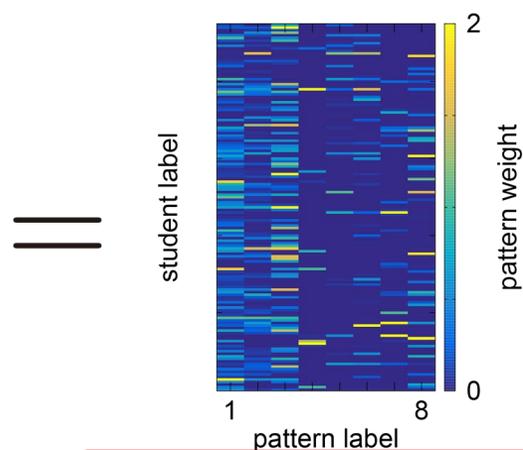
取り組みタイミングの曖昧な傾向しか見えない

→ ログ行列を2つの行列へ分解

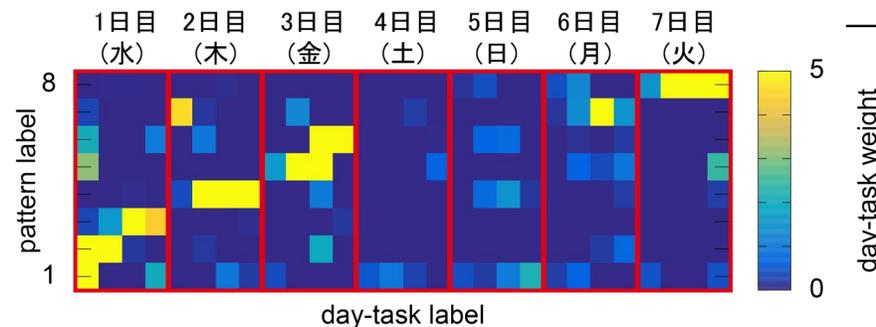
- ・パターン強度
(エラー回数の多さ)

- ・試行錯誤パターン

(e.g. 授業中 / 締め切り間際の取り組み?)



×



→ パターンの類似する学生群を抽出
(パターン強度にクラスタリング)

パターン強度

試行錯誤パターン

補:

観測行列と分解行列積の差の
フロベニウスノルムの最小化を実行

まとめ・今後の課題

授業中の進捗状況をリアルタイムに把握できる
教員の目が届かない宿題の途中経過が把握できる



ログから解析しているため小課題の回答コピペは難しい



テキストで説明されている知識から小課題を解こうとする



教員側の熱意が伝わり、大人数においても個人指導ができるようになった

課題

学生の課題への取り組み方が分類できるようになってきたが、
どのように成績に反映させるか？