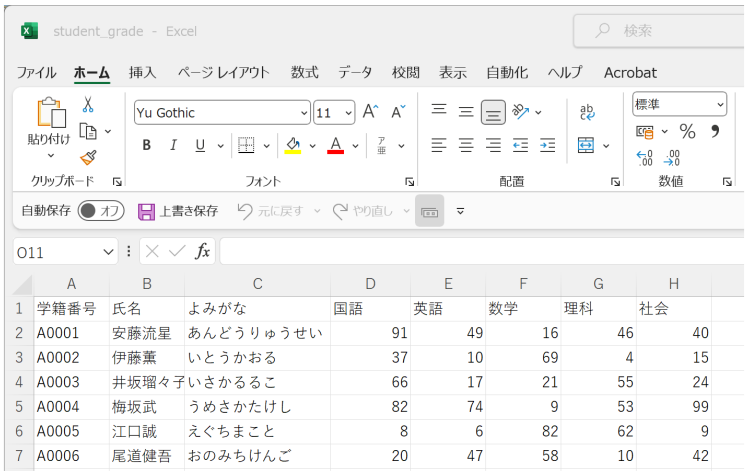
第 9 章

基盤モジュール: Pandas

表計算 (table calculation) ソフトウェアは，その名の通り，数値や文字列を含むデータを表にまとめて整理するためのデータ処理ツールです。Microsoft 社が開発した Excel（エクセル）は代表的な表計算ソフトウェアで，長い歴史を誇ります。その他，無料で使用することのできる Google Sheets や，Apache 財団の OpenOffice Calc などがあり，いずれも表を用いたデータ処理やグラフく作成機能を備えています。このように電子化された表計算ソフトウェアのデータは，あらかじめ整理されている分，Python にとって扱いやすいものと言えます。本章では主として Pandas パッケージを用いた Excel ファイルの扱い方を学び，簡単な統計ツールを作っていきます。

9.1 Excel と表

今回使用する Excel ファイルは図 9.1 に示すような，個人ごとに 5 科目（国語，英語，数学，理科，社会）の得点が記されているものとします。



	A	B	C	D	E	F	G	H
1	学籍番号	氏名	よみがな	国語	英語	数学	理科	社会
2	A0001	安藤流星	あんどうりゅうせい	91	49	16	46	40
3	A0002	伊藤薫	いとうかおる	37	10	69	4	15
4	A0003	井坂瑠々子	いさかるるこ	66	17	21	55	24
5	A0004	梅坂武	うめさかたけし	82	74	9	53	99
6	A0005	江口誠	えぐちまこと	8	6	82	62	9
7	A0006	尾道健吾	おのみちけんご	20	47	58	10	42

図 9.1　今回使用する Excel ファイルの内容

まず，Excel で簡単な集計を行ってみましょう。追加するのは

① 個人ごとの平均値 (Excel の average 関数を使用)，中央値 (median 関数)，標準偏差 (stdev.p 関数)
② 科目ごとの平均値，中央値，標準偏差をグラフ化する
③ 科目ごとに上位 5 名のリストと得点（棒グラフも含む）

とします。元のデータは別シートを作成し，そちらにすべてのデータを移して作業してみて下さい。完成版は図 9.2 のようになります。

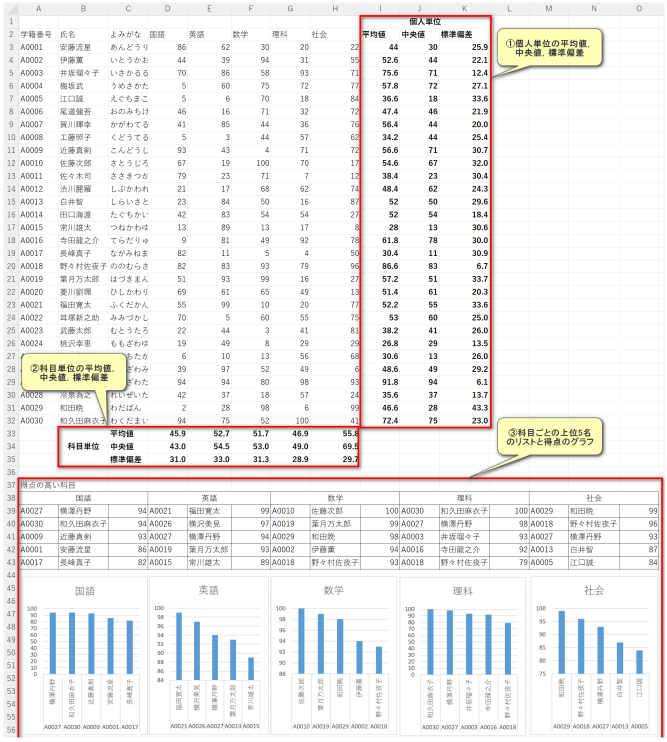


図 9.2　統計処理とグラフを付加した Excel シート

必ずしものようなレイアウトで作る必要はありませんので，自分で創意工夫したレイアウトで作成してみて下さい。ただし，次の点は気を付けて下さい。

1. 統計値はせいぜい小数点 1 桁で表示すること。
2. 得点のグラフは必ず最大値を 100 点に統一すること。

Excel だけでもこれらの処理は可能ですが，例えばこれに個人単位の得点グラフを付けろとか，個人ごとにレポート形式にせよと言われると，手作業では手間がかかります。

ということで，ここから先は Python で作業を肩代わりしてみましょう。

問題 9.1

図 9.2 を作成し，次のデータとグラフを追加せよ。

1. 科目別の最小点と最大点
2. 科目別の平均点のグラフ

9.2 Pandas を用いた Excel ファイルの読み出し

では図 9.1 に示した Excel ファイル student_grade.xlsx を Python から読み出し，図 9.2 に示すような科目別と個人別の平均値を導出してみましょう。ここで作成する Python スクリプトは「student_grade.py」とします。

9.2.1 Excel ファイルの読み出しと科目ごとの統計値算出・グラフ描画

まず pandas と japanize_matplotlib モジュールをインストールしておき，下記の部分を作成して「student_graphde.xlsx」が読み出せるか，確認して下さい。

ソースコード 9.1 student_grade.py(1/3)

```
1  # student_grade.py: 生徒の成績統計計算
2  import numpy as np # NumPy
3  import pandas as pd # Pandas
4  import matplotlib.pyplot as plt # Matplotlib
5  import japanize_matplotlib # Matplotlib 日本語対応
6
7  # Excel ファイル名
8  filename = 'student_grade.xlsx'
9  # Excel シート名
10 sheetname = '学籍番号・氏名・タイトルなし'
11 # 教科名
12 subjects = ['国語', '英語', '数学', '理科', '社会']
13
14 # Excel ファイル読み込み確認
15 try:
16     pd.ExcelFile(filename)
17 except:
18     print(filename, 'が開けませんでした。')
19
20 # Excel ファイル読み込み時のみ動作
21 with pd.ExcelFile(filename) as xls:
22     sheet = pd.read_excel(xls, sheetname)
23     print(sheet)
24
25     # 科目ごとの平均点,標準偏差,最高点,最低点
26     for i in range(len(subjects)):
27         print('{:s}の平均点,標準偏差,最高点,最低点:␣{:3.1f},␣{:5.3g},␣{:4d},␣{:4d}'.format
               (
28             subjects[i],
29             np.average(sheet[subjects[i]]),
30             np.std(sheet[subjects[i]]),
31             np.amax(sheet[subjects[i]]),
32             np.amin(sheet[subjects[i]])
33         ))
```

この部分の動作確認ができれば，科目ごとの平均点を棒グラフ化します。

ソースコード 9.2　student_grade.py(2)

```
55    # 平均点のグラフ化
56    x = subjects # ['国語', '英語', '数学', '理科', '社会']
57    y = [np.average(sheet[x[i]]) for i in range(5)]
58    fig, ax = plt.subplots()
59    # 棒グラフ
60    ax.set_title('科目別平均点')
61    ax.set_ylabel('得点')
62    ax.bar(x, y)
63    # グラフ表示
64    plt.show()
```

グラフは図 9.3 のように表示されます。

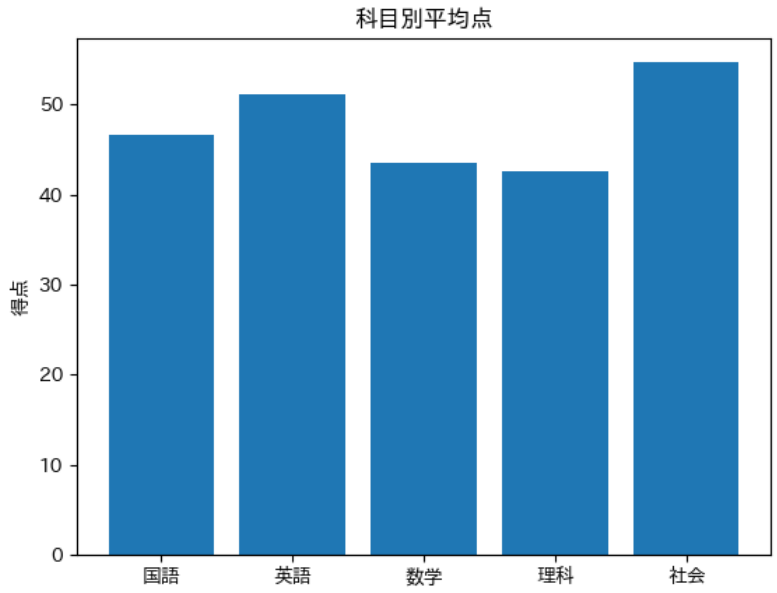


図 9.3　科目別平均点のグラフ

問題 9.2

中央値，標準偏差も導出しグラフ化せよ。

9.2.2　生徒ごとの統計値の導出

次に，生徒ごとの統計値の導出と，グラフ描画を行います。

ソースコード 9.3　student_grade.py(2)

```
63    # 生徒ごとの統計値
64    student_name = input('生徒の氏名：')
65    find_row = sheet.loc[sheet['氏名'] == student_name]
66
67    # 生徒が見つかったときのみ動作
```

```
68    if(find_row.empty == False):
69        print(student_name, '␣->␣\n', find_row)
70        #print(subjects[0], '::', find_row[subjects[0]].to_numpy()[0])
71        #print(student_name, ' -> \n', find_row.to_numpy())
72        #for i in range(len(subjects)):
73        # print(subjects[i], ' -> ', find_row[subjects[i]])
74
75        #find_row_scores = find_row.loc[student_name, subjects[0]]
76        find_row_scores = [find_row[subjects[i]].to_numpy()[0] for i in range(len(
              subjects))]
77        print(find_row_scores)
78
79        # 個人成績のグラフ化
80        x = subjects # ['国語', '英語', '数学', '理科', '社会']
81        y = find_row_scores # [find_row[x[i]] for i in range(5)]
82        #print(y)
83        fig, ax = plt.subplots()
84        # 棒グラフ
85        ax.set_title(student_name + 'の得点')
86        ax.set_ylabel('得点')
87        ax.bar(x, y)
88        # グラフ表示
89        plt.show()
90
91        print(student_name, 'の平均点,␣標準偏差,最高点,最低点:␣',
92            np.average(find_row_scores),
93            np.std(find_row_scores),
94            np.amax(find_row_scores),
95            np.amin(find_row_scores)
96        )
97
98    else:
99        print('名前:␣', student_name, '␣が見つかりません。')
```

個人別の得点グラフは図 9.4 にようになります。

問題 9.3

次の処理を追加せよ。

1. 科目ごとの上位 5 名のリストアップと棒グラフ化
2. 科目ごとの下位 5 名のリストアップと棒グラフ化

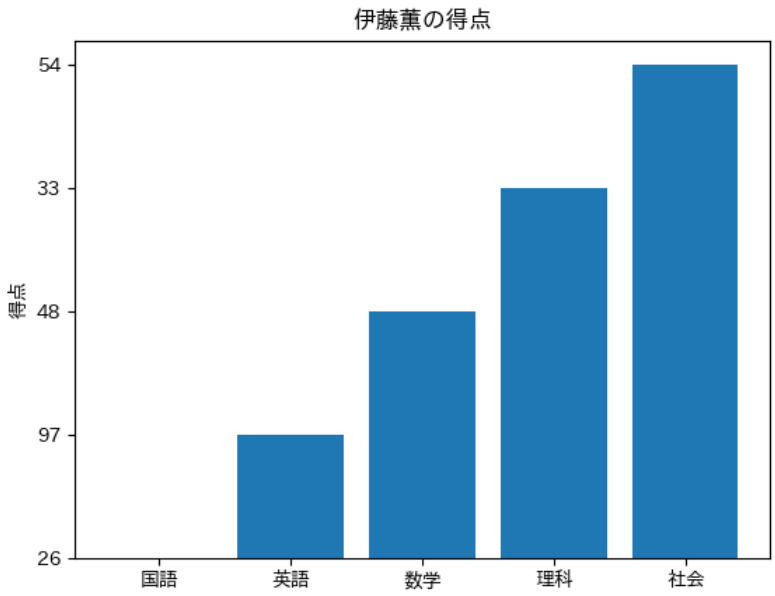


図 9.4　個人別得点グラフ