

「小数第N位まで」を答える場合は、第(N+1)位を四捨五入すること。たとえば「13.79584...」を「小数第2位まで」で答える場合は「13.80」となる。

問1. 以下の問いに答えよ。(30点)

2024-cs3-mid-1.csv を読み込んで一連の解析を行うノートブックの ____ (1) ____ などの空欄を埋め、また問いに答えよ。空欄を埋めるプログラムは1行で答えること。なお、提出するipynb/htmlファイルには、以下に表示されていないプログラム行が含まれていても構わない。

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
```

2024-cs3-mid-1.csv の全データをデータフレーム df に読み込む (1)。

```
csv_in = '2024-cs3-mid-1.csv'
____ (1) ____
```

(1)

df の行数(データ数)と列数 (2)、各列のデータ型と非欠損値数 (3)、df の最初の5行を表示 (4)。
また、データ数 (5) と列数 (6) を答えよ。

```
print( ____ (2) ____ )
print( ____ (3) ____ )
display( ____ (4) ____ )
# num of data: (5)
# num of columns: (6)
```

(2)

(3)

(4)

(5)

(6)

df から c1 列を取り出し、ser_c1 に代入 (7)。 ser_c1 の 中央値を表示 (8)。 また、その数値（小数第2位まで）を (9) に答えよ。

```
__(7)____  
print( __(8)____ )  
# value: (9)
```

(7)

(8)

(9)

dfの各行を、c1 列の値の降順にソートして、先頭5行を表示 (10)。 また、この列の2番目に大きな値（小数第2位まで）を (11) に答えよ。

```
display( __(10)____ )  
# value: (11)
```

(10)

(11)

c6 列に出現する各値とそれぞれの出現回数の一覧表示 (12)。 また、値 d の出現回数を (13) に答えよ。

```
print( __(12)____ )  
# value: (13)
```

(12)

(13)

df から ID 列、c4列、c5列を削除したデータフレーム df2 を作成 (14)。 df2 のデータを、c6 列の値ごとにまとめ、残りの列の平均値を表示 (15)。 c6 列の値が e のデータの、c2 列の平均値を (16) に答えよ(小数第2位まで)。

```
__(14)____  
display( __(15)____ )  
# value: (16)
```

(14)

(15)

(16)

df の各行について、c1 列、c2列および c3 列の値の和を求め、それを新たな n_tot 列に格納 (17)。
n_totの値について降順にソートした結果をdf3に代入 (18)。 df3の n_tot 列の先頭行の値を (19) に答えよ(小数第2位まで)。

```
__(17)____  
__(18)____  
# value: (19)
```

(17)

(18)



(19)

df3 のn_tot列のヒストグラムを作成 (20)。bin数は10とし、軸ラベルやタイトル、凡例などの装飾は
つけずに、1行で答えること。



```
__(20)____  
plt.show()
```

(20)

ipynbファイルのアップロード:

 ファイルをアップロード  未提出

htmlファイルのアップロード:

 ファイルをアップロード  未提出

* 回答は自動的に記録されますが、最後に「提出」ボタンをクリックし提出を確認してください。

 提出

「小数第N位まで」を答える場合は、第(N+1)位を四捨五入すること。たとえば「13.79584...」を「小数第2位まで」で答える場合は「13.80」となる。

問2. 以下の問いに答えよ。(25点)

cs3-mid-2-1.csv と cs3-mid-2-2.csv を読み込んで一連の処理を行うノートブックの ____ (1) ____ などの空欄を埋め、また問いに答えよ。空欄を埋めるプログラムは1行で答えること。なお、提出する ipynb/html ファイルには、以下に表示されていないプログラム行が含まれていても構わない。

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
```

2024-cs3-mid-2-1.csv の全データを **データフレーム df1**、2024-cs3-mid-2-2.csv の全データを **データフレーム df2** に読み込んであるとする(回答不要)。

df1 の重複行をすべて表示 (1)。

重複を削除して、行番号を0からの連番に振り直し、変数 df1m に代入 (2)。(元の行番号を格納する列は生成させない)

df1m の行数(データ数)を答えよ (3)。

```
display( ____ (1) ____ )
____ (2) ____
# num: (3)
```

(1)

(2)

(3)

df1m の各列の欠損値の数を表示 (4)。欠損値を1つ以上含む列をすべて答えよ (5)。次の例のように、'や'は付けず、複数ある場合はカンマで区切ること(例 a1, a2, a3)。df1m の欠損値を1つでも含む行を表示 (6)。

df1m の欠損値を1つでも含む行を削除して、行番号を0からの連番に振り直し、変数 df1m2 に代入 (7)。

```
print( ____ (4) ____ )  
# columns with missing values: (5)  
display( ____ (6) ____ )  
____ (7) ____
```

(4)

(5)

(6)

(7)

df2 の q2 列に含まれる値とその個数を表示せよ (8)。 df2 の q2 列の値't'の個数を答えよ (9)。

```
print( ____ (8) ____ )  
# number of 't': (9)
```

(8)

(9)

df1m2 の各行の右側に、df2 の対応する行を結合し、結果を変数 df3 に代入 (10)、冒頭5行を表示。このとき、df1m2 の ID 列の値が、df2 の IDX 列の値と一致するようにし、 データが ID 列と IDX 列の両方に共通して現れる場合にのみdf3に現れるようにする。

```
____ (10) ____  
display(df3.head())
```

(10)

df3の q2 列が 's'で、かつ c1 列の値が2.0より大きい行を抽出し、結果をdf3_retに代入 (11)。 df3_ret のデータフレームをCSVファイル 2024-cs3-mid-out.csv に保存 (12)。 なお、行番号をindex列に保存する必要はない。またencoding= など、その他のオプションの設定は不要。 また、df3_ret の行数 (データ数) と列数を (13) および (14) に答えよ。

```
____ (11) ____  
____ (12) ____  
# num: (13), (14)
```

(11)

(12)


(13)

(14)

ipynbファイルのアップロード:




ファイルをアップロード

 未提出

htmlファイルのアップロード:



ファイルをアップロード

 未提出

* 回答は自動的に記録されますが、最後に「提出」ボタンをクリックし提出を確認してください。



提出

「小数第N位まで」を答える場合は、第(N+1)位を四捨五入すること。たとえば「13.79584...」を「小数第2位まで」で答える場合は「13.80」となる。

問3. 以下の問いに答えよ。(25点)

データファイル「google-stock-price.csv」には、Google LLCの2020年1月1日から2023年5月15日までの株価（米ドル）が含まれている（平日のみ）。「date」列は日付を示し、「price」列は株価を示す。google-stock-price.csvを読み込んで一連の処理を行うノートブックの ____ (1) ____ などの空欄を埋め、また問いに答えよ。空欄を埋めるプログラムは1行で答えること。なお、提出する ipynb/html ファイルには、以下に表示されていないプログラム行が含まれていても構わない。

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
```

全データを**データフレーム** `df` に読み込んであるとする (回答不要)。

`df` の `date` 列を、日付を扱うのに適したデータ型に変換して上書き (1)。

```
df['date'] = ____ (1) ____
```

(1)

`df` の `'price'` 列について、オーバーラップしない1か月間ごとの平均値 (期間の始めに置く) を求め、結果を変数 `df_month_start` に格納 (2)(3)。

```
df = ____ (2) ____
df_month_start = ____ (3) ____
```

(2)

(3)

`df` に 以下のように `'remark'` 列と `'name_of_day'` 列を追加する (4)(5)。

```
df[ __(4)___ ] = ['high' if x>100 else 'low' for x in df['price']]
df[ __(5)___ ] = df.index.day_name()
```

(4)

(5)

df から 2023 年のデータを抽出し、複製メソッドを使用して、抽出されたデータのコピーを df_23 変数に代入 (6)(7)。また、df_23 の行数を(8) に答えよ。

```
df_23 = __(6)___.__(7)___
# value: (8)
```

(6)

(7)



(8)

df_23 の 'name_of_day' 列と 'remark' 列でクロス集計表を作成し(合計列・合計行も追加)、df_23_ctab に代入 (9)。



```
df_23_ctab = __(9)___
display(df_23_ctab)
```

(9)

ipynbファイルのアップロード:

 ファイルをアップロード  未提出

htmlファイルのアップロード:

 ファイルをアップロード  未提出

* 回答は自動的に記録されますが、最後に「提出」ボタンをクリックし提出を確認してください。

 提出

「小数第N位まで」を答える場合は、第(N+1)位を四捨五入すること。たとえば「13.79584...」を「小数第2位まで」で答える場合は「13.80」となる。

問4. 以下の問いに答えよ。(20点)

データファイル「driver-data.csv」には、4000人のドライバーの走行データが含まれている。

「driver-id」列はドライバーのIDを表す。「avg_dist_day」列は、ドライバーが運転した平均距離(km)を表す。「avg_over_speed」列は、ドライバーが最高速度制限を超えた平均回数を表す。

「driver-data.csv」を読み込んで一連の処理を行うノートブックの ____ (1) ____ などの空欄を埋め、また問いに答えよ。空欄を埋めるプログラムは1行で答えること。なお、提出するipynb/htmlファイルには、以下に表示されていないプログラム行が含まれていても構わない。

```
import os
os.environ['OMP_NUM_THREADS'] = '1'

import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.cluster import KMeans
from sklearn.preprocessing import scale
```

driver-data.csv の全データを**データフレームdf**に読み込んであるとする (回答不要)。

dfの「avg_dist_day」列と「avg_over_speed」列を データフレーム dfX に読み込む (1)。

dfX = ____ (1) ____

(1)

dfX を標準化し、結果を変数 X_scaled に代入 (2)。

X_scaled = ____ (2) ____

(2)

X_scaledをデータフレーム型に変換し、dfX_scaledに代入。列名はdfXと同じにする (3)。

```
dfX_scaled = ____(3)____
```

(3)

以下のように KMeans クラスタリングを実行し、結果を cls に代入。

```
kmeans = ____(4)__( ____ (5) ____ = 4, n_init=10, random_state=10)
cls = ____(6)__.____(7)__(dfX_scaled)
```

(4)

(5)

(6)

(7)

これらのクラスタのうち、クラスタ2 の 中心の座標を表示する (8)。

```
print(____(8)____)
```

(8)



クラスタリング結果を df に、「clstr_num」というラベルを持つ新しい列として追加 (9)。また、各クラスタのメンバー数を求めるコードを答えよ (10)。

```
____(9)____
display(df.head())
print(____(10)____)
```



(9)

(10)

ipynbファイルのアップロード:

 ファイルをアップロード  未提出

htmlファイルのアップロード:

 ファイルをアップロード  未提出

* 回答は自動的に記録されますが、最後に「提出」ボタンをクリックし提出を確認してください。

 提出