

```
import requests
from io import StringIO
import string
import pandas as pd
import numpy as np
```

```
import matplotlib.pyplot as plt
from mpl_toolkits.mplot3d import Axes3D
```

```
#地形データを読み込む
```

```
url = 'http://cyberjapandata.gsi.go.jp/xyz/dem/13/7228/3208.txt'
response = requests.get(url)
```

```
#標高値がない区画はとりあえず0mに置換する
```

```
maptxt = response.text.replace(u'e', u'-0.0')
```

```
#maptxt = string.replace(response.text, u'e', u'-0.0')
```

```
Z = pd.read_csv(StringIO(maptxt), header=None)
```

```
#X,Y軸のグリッドを生成
```

```
X, Y = np.meshgrid(np.linspace(0,255,256), np.linspace(255,0,256))
```

```
fig = plt.figure(figsize=(13, 10), dpi=80, facecolor='w', edgecolor='k')
```

#3Dグラフ

```
# プロット中の軸の取得。gca はGet Current Axes の略。
```

```
ax = fig.gca(projection='3d')
```

```
# 横を1/0.8=1.25倍長く設定
```

```
ax.set_aspect('auto', adjustable='box')
```

```
# ax.set_aspect(0.8, adjustable='box')
```

```
# 上高地の遙か上空ぐらいから前穂高越しに地形を見下ろす感じに視点を設定
```

```
ax.view_init(70, -67)
```

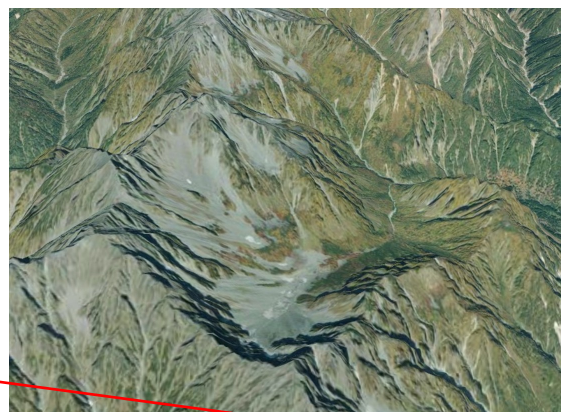
```
#wireframeを描く
```

```
ax.plot_wireframe(X, Y, Z, rstride=1, cstride=2, linewidth=1)
```

```
#plt.savefig('map.jpg', dpi=72)
```

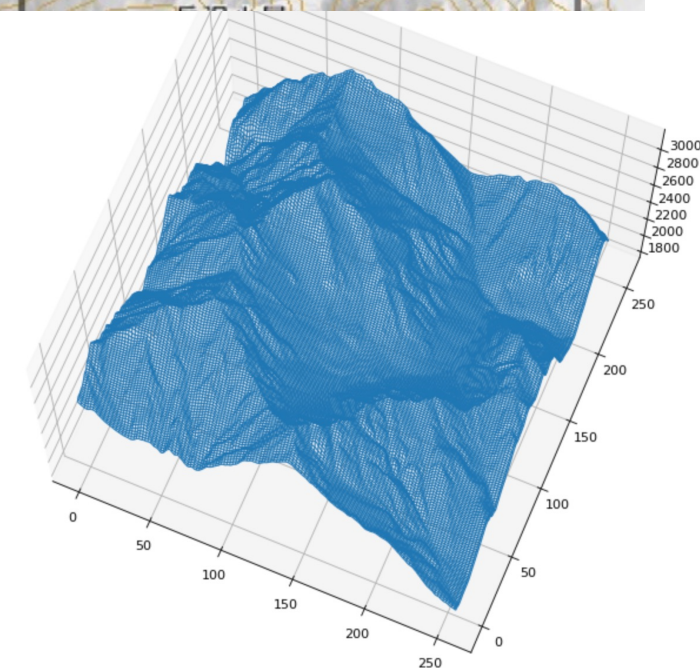
```
plt.show()
```

出典: <https://memomemokun.hateblo.jp/entry/2018/11/02/093839>



replace メソッドは、文字列内の指定された部分文字列を別の文字列に置き換えるために使用されます。このメソッドは、str クラスのメソッドとして提供されているため、string モジュールからインポートする必要はありません

aspect パラメータは、3D プロットの縦横奥の比率を制御します。auto に設定すると、データの範囲に自動的にフィットするように比率が調整されます。



```

import requests
from io import StringIO
import string
import pandas as pd
import numpy as np

import matplotlib.pyplot as plt
from mpl_toolkits.mplot3d import Axes3D

#地形データを読み込む
url = 'http://cyberjapandata.gsi.go.jp/xyz/dem/13/7228/3208.txt'
response = requests.get(url)

#標高値がない区画はとりあえず0mに置換する
maptxt = response.text.replace(u'e', u'-0.0')
#maptxt = string.replace(response.text, u'e', u'-0.0')
Z = pd.read_csv(StringIO(maptxt), header=None)

#X,Y軸のグリッドを生成
X, Y = np.meshgrid(np.linspace(0,255,256), np.linspace(255,0,256))

fig = plt.figure(figsize=(14, 10), dpi=80, facecolor='w', edgecolor='k')

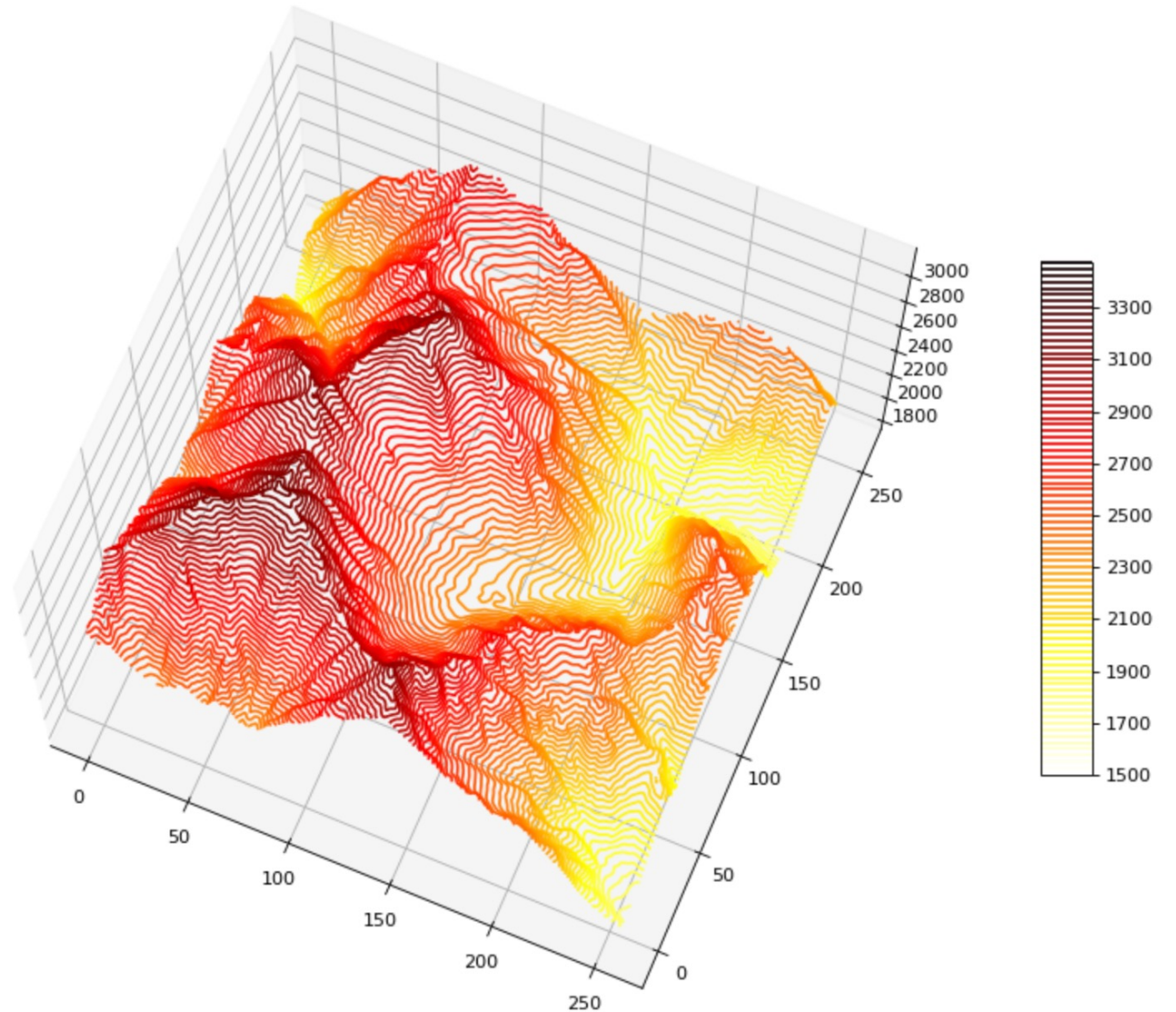
#プロット中の軸の取得。gca はGet Current Axes の略。
ax = fig.gca(projection='3d')
#横を1/0.8=1.25倍長く設定
ax.set_aspect('auto', adjustable='box')
#ax.set_aspect(0.8, adjustable='box')
#上高地の遙か上空ぐらいから前穂高越しに地形を見下ろす感じに視点を設定
ax.view_init(70, -67)

#標高25m間隔で等高線を描く
elevation = range(1500,3500,25)
cont = plt.contour(X, Y, Z, levels=elevation, cmap='hot_r') #等高線

#ラベルをつける
cb = plt.colorbar(cont, shrink=0.5, aspect=10)

#plt.savefig('map.jpg', dpi=72)
plt.show()

```



国土地理院のサイトでタイルを選択してデータをダウンロード

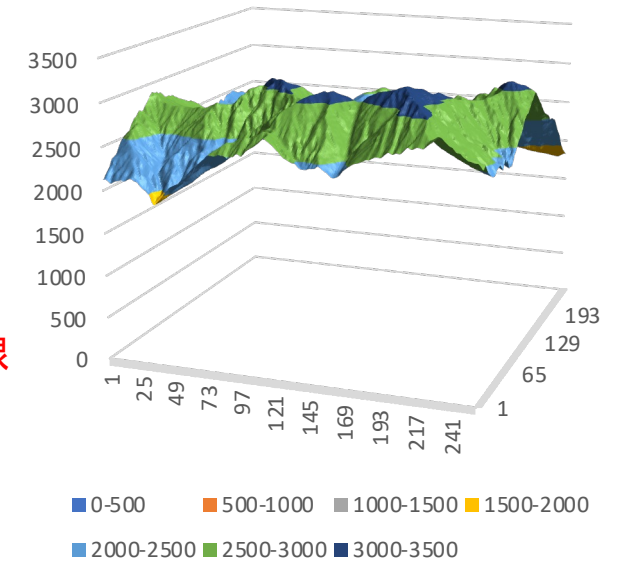
<http://cyberjapandata.gsi.go.jp/xyz/dem/13/7228/3208.txt>

標高タイル
テキスト(単位m)
256個

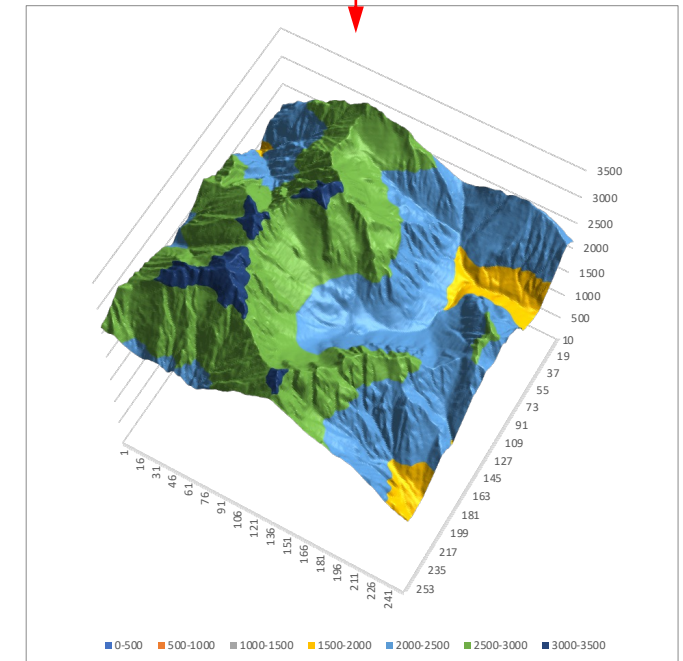
256行



Excelの等高線でプロット



3D回転



pixel座標の標高値
標高値がない場合は「e」

```
2121.15,2126.00,2135.03,2152.35,2167.84,2182.77,2194.69,2204.43,2213.70,2226.85,2242.33,2264.
2454.25,2469.60,2482.87,2494.53,2503.55,2510.13,2516.09,2517.68,2523.52,2531.00,2533.40,2533.
2704.64,2717.21,2728.73,2737.78,2733.64,2721.10,2717.01,2715.50,2714.88,2723.74,2735.95,2754.
2868.89,2869.57,2870.81,2870.53,2873.98,2869.86,2854.08,2842.85,2849.22,2851.99,2850.32,2837.
2756.80,2757.39,2760.08,2759.92,2753.00,2745.35,2735.42,2727.95,2718.75,2705.31,2686.09,2658.
2520.93,2509.65,2496.79,2483.03,2473.53,2465.09,2455.47,2445.36,2435.99,2427.22,2419.05,2410.
2283.13,2277.31,2273.34,2271.21,2270.47,2272.29,2273.65,2274.36,2278.37,2283.31,2287.24,2291.
2389.72,2393.99,2397.80,2398.07,2402.12,2410.78,2420.35,2431.53,2440.52,2444.50,2438.70,2425.
2472.22,2477.49,2483.55,2487.19,2493.81,2496.49,2493.58,2488.43,2479.81,2470.60,2461.69,2452.
2366.97,2359.03,2349.88,2340.82,2330.95,2317.36,2303.98,2291.27,2282.64,2271.30,2257.76,2244.
2128.56,2135.81,2146.80,2161.56,2175.83,2190.25,2203.42,2214.47,2226.52,2235.63,2247.55,2271.
2463.22,2479.36,2495.04,2510.32,2520.44,2525.99,2532.15,2534.80,2538.64,2547.05,2550.32,2551.
2701.61,2710.72,2718.82,2728.30,2724.29,2708.59,2703.23,2700.01,2706.86,2721.44,2739.31,2751.
2858.67,2856.00,2856.31,2855.46,2853.32,2846.23,2834.42,2831.61,2836.61,2835.58,2834.72,2828.
2739.62,2740.91,2742.68,2741.37,2737.44,2729.34,2720.90,2718.69,2717.28,2704.43,2684.95,2660.
2533.11,2522.64,2508.66,2493.32,2482.05,2473.06,2464.16,2453.86,2443.50,2433.99,2425.64,2417.
2285.32,2276.82,2270.23,2266.21,2265.14,2266.50,2267.93,2268.76,2271.87,2276.47,2280.20,2283.
2379.55,2383.09,2383.10,2385.60,2394.29,2404.96,2413.95,2422.33,2427.36,2427.68,2424.77,2414.
2458.66,2464.23,2472.01,2475.39,2478.27,2479.33,2478.52,2477.60,2474.06,2468.59,2463.63,2459.
2382.65,2376.52,2368.30,2359.82,2351.49,2337.71,2326.87,2312.39,2295.68,2283.51,2269.33,2255.
2129.36,2137.63,2148.89,2161.28,2172.39,2185.11,2197.89,2209.93,2223.89,2235.39,2253.14,2283.
2470.78,2486.00,2503.61,2523.34,2535.96,2542.31,2548.78,2553.14,2555.82,2560.73,2564.80,2567.
2694.12,2700.93,2708.37,2718.76,2715.88,2699.63,2688.66,2690.21,2705.01,2721.37,2735.63,2739.
2848.61,2843.30,2843.25,2840.27,2838.25,2832.43,2822.16,2815.59,2814.31,2809.67,2809.10,2810.
2724.70,2723.32,2725.98,2724.85,2720.18,2712.00,2705.54,2706.05,2710.46,2701.98,2683.40,2662.
2547.10,2535.52,2522.00,2507.00,2492.91,2482.07,2472.47,2462.43,2451.80,2441.58,2432.01,2423.
2287.32,2278.11,2269.48,2262.13,2258.43,2259.91,2262.52,2263.61,2265.79,2269.18,2272.43,2275.
2366.35,2369.58,2369.75,2378.16,2388.43,2397.55,2405.56,2412.89,2414.76,2415.31,2413.32,2403.
2445.94,2450.48,2457.36,2459.75,2459.97,2460.82,2461.77,2462.47,2461.24,2458.39,2454.06,2450.
2396.65,2391.29,2385.77,2378.94,2368.74,2357.96,2347.61,2334.82,2314.10,2296.38,2282.06,2267.
```

256pixel

