

# 色を自在に操る (HSVカラーコードのすすめ)

関根 暁史

株式会社 ACRONET / 生物統計部

How to use a kind of color codes -The HSV color codes makes colors easily-

Satoshi Sekine

ACRONET Corp./Biostatistics Dept. Data Science Division

## 要旨

SAS に装備されているカラーコードは、カラーネームにない色を自在に定義出来、グラフなどに使用することができる。そのカラーコードのプログラム中での使用方法を紹介する。

キーワード : RGB, CMYK, HSV, HLS, Gray-Scale, 16 進数

## 1. はじめに

度数の大小を色の濃淡で表現した図を描こうと考えた時、カラーネームにおける very light, light, medium, dark, very dark の修飾語を用いて表現しても 5 段階、更に white, black を追加したとしても 7 段階までの表現が限界である。しかしユーザーがカラーコード(RGB, CMYK, HSV, HLS, Gray-Scale)を覚えた場合、ほぼ自在の表現が可能となろう。本発表では上記 5 種のカラーコードの概念とそれぞれの相互関係について概説し、特に色彩学の知識がなくとも色の定量性について理解のし易い HSV カラーコードのプロシジャなどでの具体的活用例を紹介する。

## 2. カラーコードの定義

<RGB>

RGB とは R : 赤, G : 緑, B : 青の加法混色の表現法であり、ディスプレイやデジタルカメラなどに使用されているカラーモデルである。SAS での RGB カラーコードの指定方法は下記の通りである。なお 10 進数と 16 進数の相互変換は hex.フォーマットを介することで簡単に行える。

**CX00FF00**  
① ② ③ ④

- ① RGB カラーコードを示す。
  - ② 0~255(256 段階)の R を 16 進数化した 2 桁を入力する。
  - ③ 0~255(256 段階)の G を 16 進数化した 2 桁を入力する。
  - ④ 0~255(256 段階)の B を 16 進数化した 2 桁を入力する。
- この例では green を表している。

## <CMYK>

CMY とは C：シアン，M：マゼンタ，Y：黄の減法混色の表現法である。シアンは赤と、マゼンタは緑と、黄は青と互いに補色(反対色)の関係である。補色とは補数の関係でもある。CMYK は CMY から派生した表現法であり、純粋な黒色を塗料の調合で表現するのが困難であることから K：黒を追加したものである。CMYK はプリンターや塗料の配色に使用されているカラーモデルである。SAS での CMYK カラーコードの指定方法は下記の通りである。

0075294F

① ② ③ ④

- ①0～255(256段階)の C を 16 進数化した 2 桁を入力する。
- ②0～255(256段階)の M を 16 進数化した 2 桁を入力する。
- ③0～255(256段階)の Y を 16 進数化した 2 桁を入力する。
- ④0～255(256段階)の K を 16 進数化した 2 桁を入力する (CMY カラーコードとして用いたい場合は K='00' としておけばよい)。

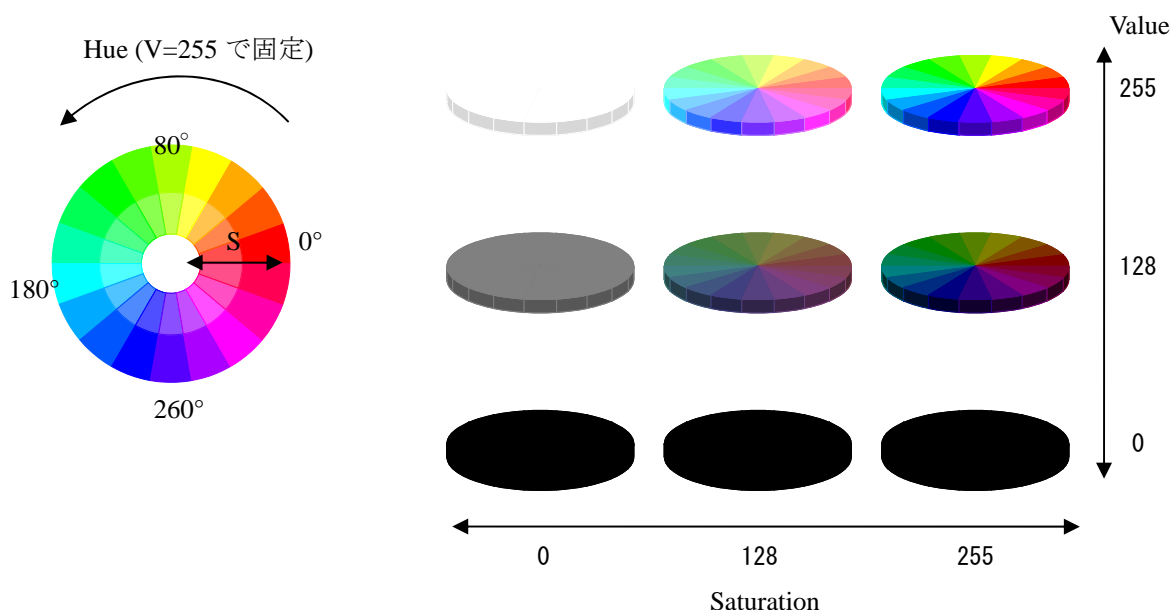
※接頭語は不要。

この例では purple を表している。

## <HSV>

RGB で色を自在に調整できることは判ったが、R と G と B のそれぞれをどの程度混色したら何色になるかという事は直感的に判りにくいものがある (青と赤を混色したら紫になることは何となく判っても、それ以外の色の配色が何色になるかが判りにくい)。そこで RGB 色空間を人間が認識し易いものに変換したものが HSV 色空間である。RGB⇔HSV の相互変換については後述する。HSV は H：Hue(色相)、S：Saturation(彩度)、V：Value(明度)からなるもので、360 度の色相環があり、彩度が下がるほど灰色(無彩色)に近づき、明度が下がるほど暗くなるという概念である。HSV の概念図を図 0-1 に示す。

図 0-1. HSV 色空間の概念図



また、SAS での HSV カラーコードの指定方法は下記の通りである。

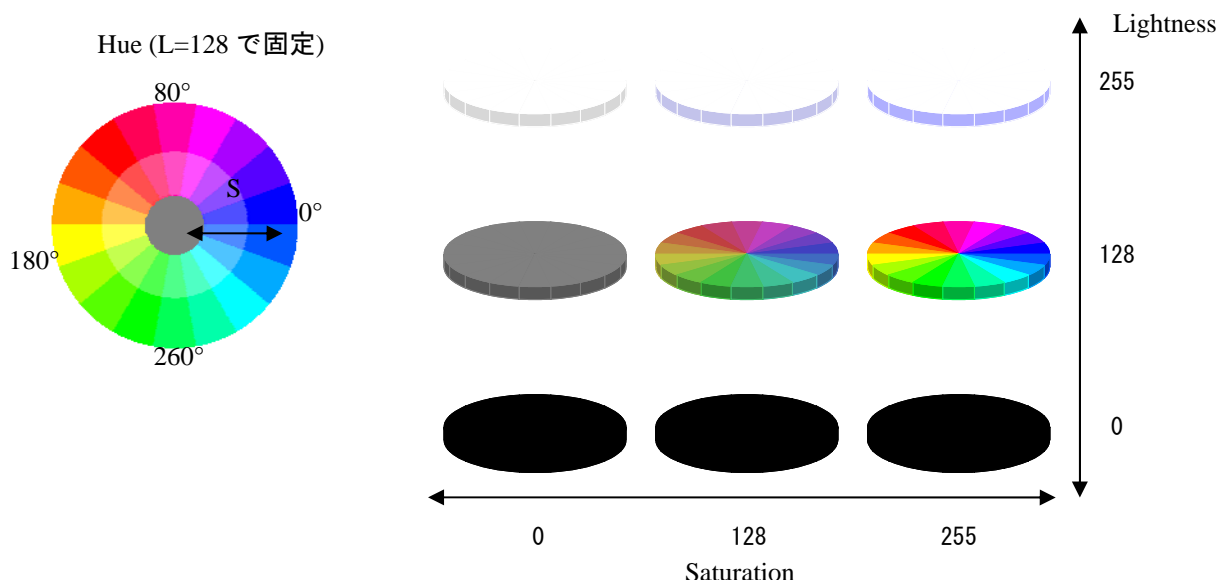
**V000FFBF**  
 ① ② ③ ④

- ①HSV カラーコードを示す。
  - ②0~359(360 段階)の H を 16 進数化した 3 桁を入力する。
  - ③0~255(256 段階)の S を 16 進数化した 2 桁を入力する。
  - ④0~255(256 段階)の V を 16 進数化した 2 桁を入力する。
- この例では dark red を表している。

### <HLS>

HLS は H : Hue(色相)、L : Lightness(輝度)、S : Saturation (彩度)からなるものもので、HSV とよく類似したカラーモデルであるが、RGB からの変換公式が HSV とは若干異なるものである(輝度が 128 を超えると白に近づいていく)。HLS の概念図を図 0-2 に示すが、(SAS では)HSV とは色相環が 240 度ずれているので注意されたい。

図 0-2. HLS 色空間の概念図



SAS には色に関する様々な関数が整備されており、RGB⇔HLS の相互変換例と併せて解説する。

```
/* 相互変換例 */
%COLORMAC;
data HENKAN;
  HLS="%CNS(green)";
  RGB="%HLS2RGB(H0F080FF)";
  Re_HLS="%RGB2HLS(CX00FF00)";
run;
```

%COLORMAC;と書けば色に関する関数を呼び出せる。例えば SAS において green がどのような HLS コードに相当するか知りたい時、%CNS を使えば H0F080FF であることが判る。H0F080FF を%HLS2RGB にかければ、RGB コードでは CX00FF00 に相当することが判る。CX00FF00 を%RGB2HLS にかければ H0F080FF に戻る。

また、SAS での HLS カラーコードの指定方法は下記の通りである。

# H00080FF

① ② ③ ④

- ①HLS カラーコードを示す。
- ②0~359(360 段階)の H を 16 進数化した 3 桁を入力する。
- ③0~255(256 段階)の L を 16 進数化した 2 桁を入力する。
- ④0~255(256 段階)の S を 16 進数化した 2 桁を入力する。

この例では blue を表している。

## <Gray-Scale>

Gray-Scale はモノクロのカラーコードである。SAS での Gray-Scale カラーコードの指定方法は下記の通りである。

# GRAY00

① ②

- ①Gray-Scale カラーコードを示す。
  - ②0~255(256 段階)の GRAY の値を入力する。
- この例では black を表している。

## <カラーコード間の相互関係>

上記 5 種のカラーコード間の相互関係を、グレーのチャートを通して概説する。図 0-3 のコードは全て左のグレーのチャートに相当している。

図 0-3. 各コードの相互関係

グレーの チャート	①:RGB	②:CMY	③:CMYK	④:HSV	⑤:HLS	⑥:Gray-Scale
	CX000000	FFFFFF00	000000FF	Vxxx0000	Hxxx0000	GRAY00
	CX404040	BFBFBF00	000000BF	Vxxx0040	Hxxx4000	GRAY40
	CX808080	7F7F7F00	0000007F	Vxxx0080	Hxxx8000	GRAY80
	CXC0C0C0	3F3F3F00	0000003F	Vxxx00C0	HxxxC000	GRAYC0
	CXFFFFFF	00000000	00000000	Vxxx00FF	HxxxFF00	GRAYFF

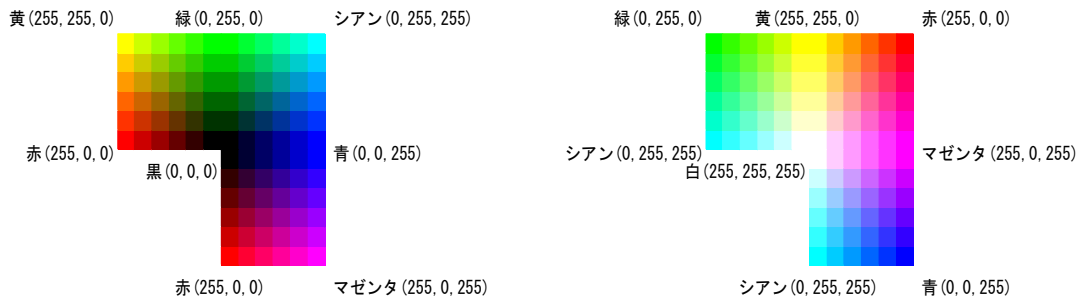
x は何を入力してもよい

- ①：無彩色のとき R=G=B である。値が大きいくほど白くなる(加法混色)。
- ②：RGB とは補数の関係である。値が大きいくほど黒くなる(減法混色)。
- ③：無彩色のとき②の CMY の値を K に代入し、CMY は'00'としたものが相当する。
- ④：無彩色のとき S='00'であり、V の値は①に対応している。
- ⑤：無彩色のとき S='00'であり、L の値は①に対応している。
- ⑥：値は①(または④, ⑤)に対応している。

## <RGB・HSV 相互関係>

RGB 色空間とは、下図 0-4 の左右の図を切り取って組み合わせて作った立方体のようなものである。RGB=(0,0,0)のとき完全な黒であり、RGB=(255,255,255)のとき完全な白である。この立方体の中心には灰色(無彩色)が存在している。

図 0-4. RGB カラーチャート



RGB⇄HSV の相互変換を行う関数は SAS に装備されていないため、他言語などのプログラムを参考に自作してみた。green に相当する RGB コード CX00FF00 を用いた例を下に紹介する。RGB コード CX00FF00 は HSV では V078FFFF に変換される。

```
/* RGB から HSV への変換 */
```

```
data RGBtoHSV;
```

```
  RGB="CX00FF00";
```

```
  R=input(substr(RGB,3,2),hex.);
```

```
  G=input(substr(RGB,5,2),hex.);
```

```
  B=input(substr(RGB,7,2),hex.);
```

```
  MAX=max(R,G,B);
```

```
  MIN=min(R,G,B);
```

```
  if MAX=0 then do;
```

```
    H=0; S=0; V=0;
```

```
  end;
```

```
  else if MAX=MIN then do;
```

```
    H=0; S=0; V=MAX;
```

```
  end;
```

```
  else do;
```

```
    if MAX=R then
```

```
      H=mod(60*(G-B)/(MAX-MIN)+360,360);
```

```
        if MAX=G then H=60*(B-R)/(MAX-MIN)+120;
```

```
        if MAX=B then H=60*(R-G)/(MAX-MIN)+240;
```

```
        if MAX=0 then S=0;
```

```
        else S=255*((MAX-MIN)/MAX);
```

```
        V=MAX;
```

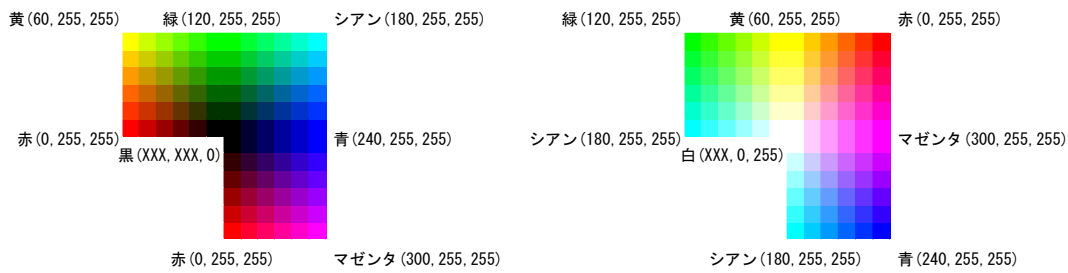
```
      end;
```

```
      HSV="V"||put(round(H,1),hex3.)||put(round(S,1),hex2.)||put(round(V,1),hex2.);
```

```
    run;
```

上記プログラムを用いて RGB から HSV に変換したコードで再作成したカラーチャートを。図 0-4 に示す。図 0-4 と図 0-5 のラベル以外のカラーチャート部分は PDF 比較によって同一のものであることを確認している。図 0-5 において赤(0)→黄(60)→緑(120)→シアン(180)→青(240)→マゼンタ(300)→赤(0)とたどると、360 度の色相環になっているのがお判り頂けると思う。

図 0-5. (RGB から変換した)HSV カラーチャート



※図中の X は何が入ってもよい

念のため HSV から RGB への変換プログラムも左に示す。先ほどの green に相当する HSV コード V078FFFF は RGB コード CX00FF00 に変換される。

```

/* HSV から RGB への変換 */
data HSVtoRGB;
  HSV="V078FFFF";
  H=input(substr(HSV,2,3),hex.);
  S=input(substr(HSV,5,2),hex.);
  V=input(substr(HSV,7,2),hex.);
  Hi=mod(floor(H/60),6);
  f=H/60-floor(H/60);
  p=round(V*(1-(S/255)),1);
  q=round(V*(1-(f*S/255)),1);
  t=round(V*(1-((1-f)*S/255)),1);
  if Hi=0 then do; R=V; G=t; B=p; end;
  if Hi=1 then do; R=q; G=V; B=p; end;
  if Hi=2 then do; R=p; G=V; B=t; end;
  if Hi=3 then do; R=p; G=q; B=V; end;
  if Hi=4 then do; R=t; G=p; B=V; end;
  if Hi=5 then do; R=V; G=p; B=q; end;
  RGB="CX"||put(R,hex2.)||put(G,hex2.)||put(B,hex2.);
run;

```

以上はカラーコードの基本概念を述べさせて頂いたが、次章からは HSV を利用したカラーコードの具体的な応用例を紹介させて頂く。

### 3. 平面塗り分け地図の作成

この章では平面塗り分け地図の作成を通して、データ値の持つ定量性を色彩で表現することを試みる。黄色を基調として、人口密度(人/平方 km)が 1000 上がるごとに赤に近づく大阪府各市区町村別の人口密度マップを作成した。図 1-1 は凡例を頼らずとも大阪府の人口の重心がどこにあるかを知ることが出来る図である。紙面の都合上図 1-1 の地図部分のみの作成プログラムを記す。

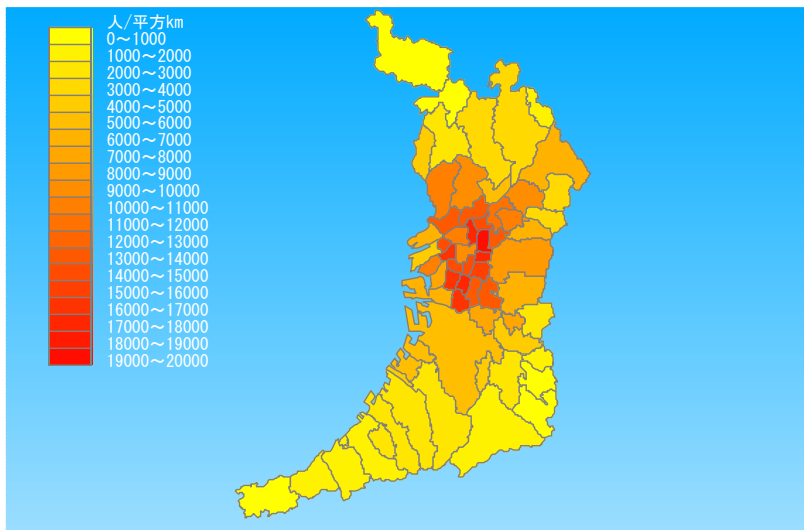


図 1-1. 大阪府の各市区町村別の人口密度

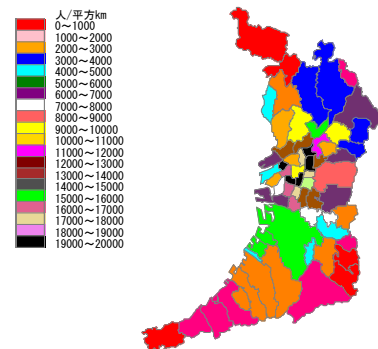


図 1-2

```
/* 図 1-1 地図部分塗り分けプログラム */  
/* SAS9.2 の MAPS を使用 */  
data PATTERN; set MAPS.JAOSAKA2;  
  if ID in(118) then val=3;  
  if ID in(102,115,119) then val=9;  
  if ID in(106,120,122) then val=12;  
  if ID in(116) then val=15;  
  if ID in(103,109,111,117) then val=18;  
  if ID in(114,121,123,124,126) then val=21;  
  if ID in(209) then val=27;  
  if ID in(101,107,203,223) then val=30;  
  if ID in(105,205,215) then val=33;  
  if ID in(227) then val=36;  
  if ID in(108,217,226) then val=39;  
  if ID in(113,125,210,212,218) then val=42;  
  if ID in(201,206,224,225) then val=45;
```

```
  if ID in(104,204,222,231,341) then val=48;  
  if ID in(207,211,229,230) then val=51;  
  if ID in(202,208,214,219,220,221,361) then val=54;  
  if ID in(213,216,228,301,362,367,381) then val=57;  
  if ID in(321,322,366,382,383) then val=60;  
  HSV="V"||put(val,hex3.)||"FFFF";  
  PATTERN="pattern"||left(put(_n_,best.));  
run;  
data _NULL_; set PATTERN;  
  call execute(PATTERN||" v=s c="||HSV||");  
run;  
proc gmap data=MAPS.JAOSAKA2  
  map=MAPS.JAOSAKA ;  
  id ID ; choro ID / discrete nolegend  
  outline=V0000080 ;  
run; quit;
```

もしカラーコードを知らずに、有り合わせのカラーネームを使用して地図を作ると図 1-2 のようになってしまうかも知れない。図 1-2 では人口密度の定量感が掴みにくい。

## 4. ヒストグラムの作成

この章ではヒストグラムの作成を通して、学会発表等のプレゼンテーションにも耐えうる品位の SAS グラフの作成を試みる。図 2-1 は SAS によるデータ解析入門[第 2 版] (東京大学出版会)p.96~に掲載のデータ height を GCHART プロシジャに供した時の SAS9.2 のデフォルトの結果である。このデータを元に GPLOT プロシジャと ANNOTATE データセットの組み合わせで色彩豊かに作成したヒストグラムが図 2-2 である。

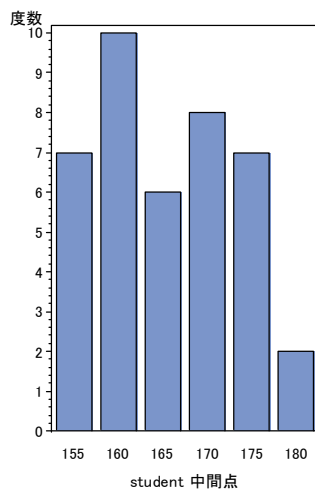


図 2-1

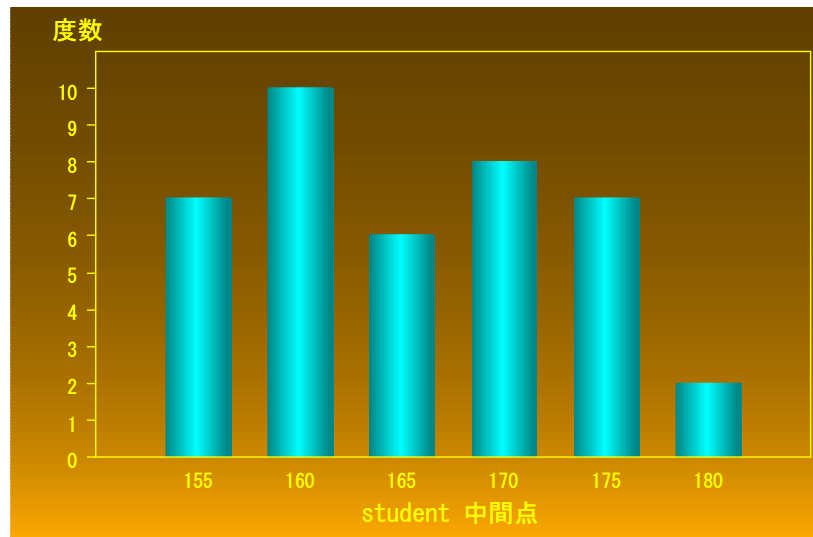


図 2-2

```
/* 図 2-2 BAR 部分描画プログラム */
data height;
  set height;
  if student<157.5 then x_cat=10;
  if 157.5<=student<162.5 then x_cat=20;
  if 162.5<=student<167.5 then x_cat=30;
  if 167.5<=student<172.5 then x_cat=40;
  if 172.5<=student<177.5 then x_cat=50;
  if 177.5<=student then x_cat=60;
run;
proc freq data=height noprint;
  table x_cat/out=table;
run;
data x_dat; set table; low=x_cat-3; hi=x_cat+3;
  do i=low to hi by 0.1; output; end;
run;
%annomac;
```

```
data x_dat; set x_dat;
  if round(i-x_cat,1e-8)>0 then
val=255-40*round(i-x_cat,1e-8);
  else val=255+40*round(i-x_cat,1e-8);
  lower=i-0.1; upper=i+0.1;
  HSV="V0B4FF"||put(val,hex2.);
  length TEXT $80. ; %dclanno; %system(2, 2, 2 );
  %BAR(lower,0,upper,count,color,1,s);
run;
data x_dat; set x_dat; color=HSV; run;
proc gplot data=x_dat anno=x_dat;
  plot COUNT*x_cat
  / nolegend noframe noaxes vaxis=0 to 11 by 1 haxis=0
to 70 by 10;
  symbol v=none;
run;
quit;
```

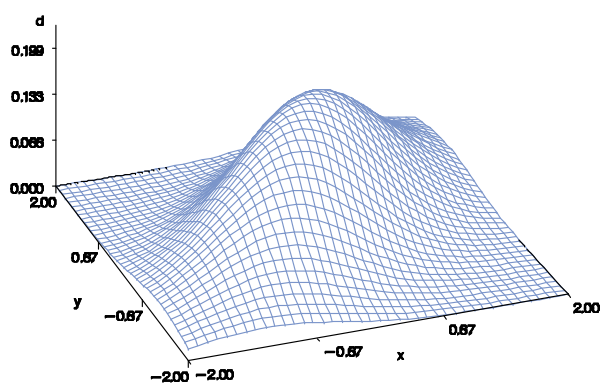


紙面の都合上 BAR 部分のみ描画プログラムを記載したが、具体的にはシアンを基調として中心から左右に離れるほど 30 段階程度明度を落とすことで BAR を円柱的に見せている。さらに背景は茶色を基調とした 150 段階程度のシームレスなグラデーションをかけている。図 2-2 の品位であれば、そのままパワーポイントにも貼れそうである。

## 5. 三次元プロットの作成

この章では三次元の surface プロットに色づけすることを試みる。

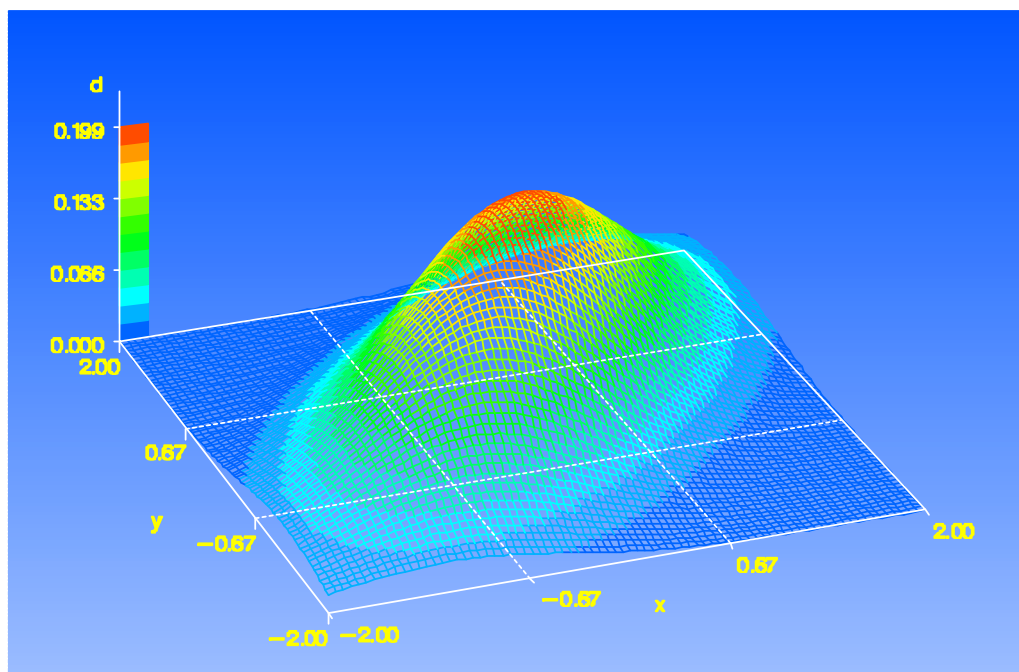
図 3-1. 二次元正規分布



左図 3-1 は SAS によるデータ解析入門[第 2 版] (東京大学出版会)p.135～に掲載のデータ normal を G3D プロシジャに供した時の SAS9.2 のデフォルトの結果である。残念ながら G3D プロシジャの surface プロットは単色の指定しかできないため、山の高さが目盛のどの位置にあるか掴みづらい。そこで Z 軸の高さに応じて山を塗り分けし(12 段階)、凡例を追加することで等高レベルを明らかにしてみる。塗り分けの方法は SAS CUSTOMER SUPPORT に記載(Sample 24861)の方法を採用した

([http://support.sas.com/sassamples/graphgallery/PROC\\_G3D.html](http://support.sas.com/sassamples/graphgallery/PROC_G3D.html))。

図 3-2. 二次元正規分布



```

/* 図 3-2 山部分描画プログラム (論文集掲載のもの
より細かく描画します) */
data normal;
  r=0.6; pai=3.141593;
  c=(1/(2*pi*(1-r**2)**0.5));
  do x=-2 to 2 by 0.05;
    do y=-2 to 2 by 0.05;
      d=c*exp(-(0.5/(1-r**2)*(x**2-2*r*x+y**2)));
      output;
    end;
  end;
run;
%let nlevels=12;
%let colors="V0D8FFFF V0C6FFFF V0B4FFFF
V0A2FFFF V090FFFF V07EFFFF V06CFFFF
V05AFFFF V048FFFF V036FFFF V024FFFF
V012FFFF";
data normal; set normal;
  _z=d; _y=y; _x=x;
run;
proc means data=normal noprint min max;
  var _x _y _z;
  output out=range
    min=_xmin _ymin _zmin
    max=_xmax _ymax _zmax;
run;
data _null_;
  set range;
  call symput('_xmin', _xmin);
  call symput('_xmax', _xmax);
  call symput('_ymin', _ymin);
  call symput('_ymax', _ymax);
  call symput('_zmin', _zmin);
  call symput('_zmax', _zmax);
  call symput('floor', 0);
  call symput('ceil', 0.199);
  call symput('step', (_zmax-_zmin)/&nlevels);
run;

```

```

proc sort data=normal;
  by _x;
run;
data surf1;
  length function color $8;
  retain xsys ysys zsys '2';
  drop _y _x _z ncol;
  set normal(drop=x y d);
  by _x;
  x=_x; y=_y; z=&floor;
  if first._x then function='move';
  else
    do;
      function='draw';
      ncol=min(&nlevels,int(1+round(_z-&_zmin,1e-8)/&step
)); color=scan(&colors,ncol);
    end;
  z=_z; output surf1;
run;
proc sort data=normal;
  by descending _y _x;
run;
data surf2;
  length function color $8;
  retain xsys ysys zsys '2';
  drop _y _x _z ncol;
  set normal(drop=x y d);
  by notsorted _y;
  x=_x; y=_y; z=&floor;
  if first._y then function='move';
  else
    do;
      function='draw';
      ncol=min(&nlevels,int(1+round(_z-&_zmin,1e-8)/&step
)); color=scan(&colors,ncol," ");
    end;
  z=_z; output surf2;
run;

```

```

data annoall;
  set surf2 surf1 ;
run;
data plotdata;
  x=&_xmin; y=&_ymin; d=&floor; output;
  x=&_xmax; y=&_ymax; output;
run;
proc g3d data=plotdata ;
  scatter y*x=d/shape="point" rotate=20 tilt=40 zmax=&ceil annotate=annoall;
run; quit;

```

図 3 シリーズは相関係数=0.6 の時の二次元正規分布であるが、図 3-2 では等高レベルが明らかになっただけでなく、等高線が楕円形状であることも見てとれる。さらに山の輪郭線も強調できている。但し外挿用の ANNOTATE データセットの作成には手間がかかるだろう。

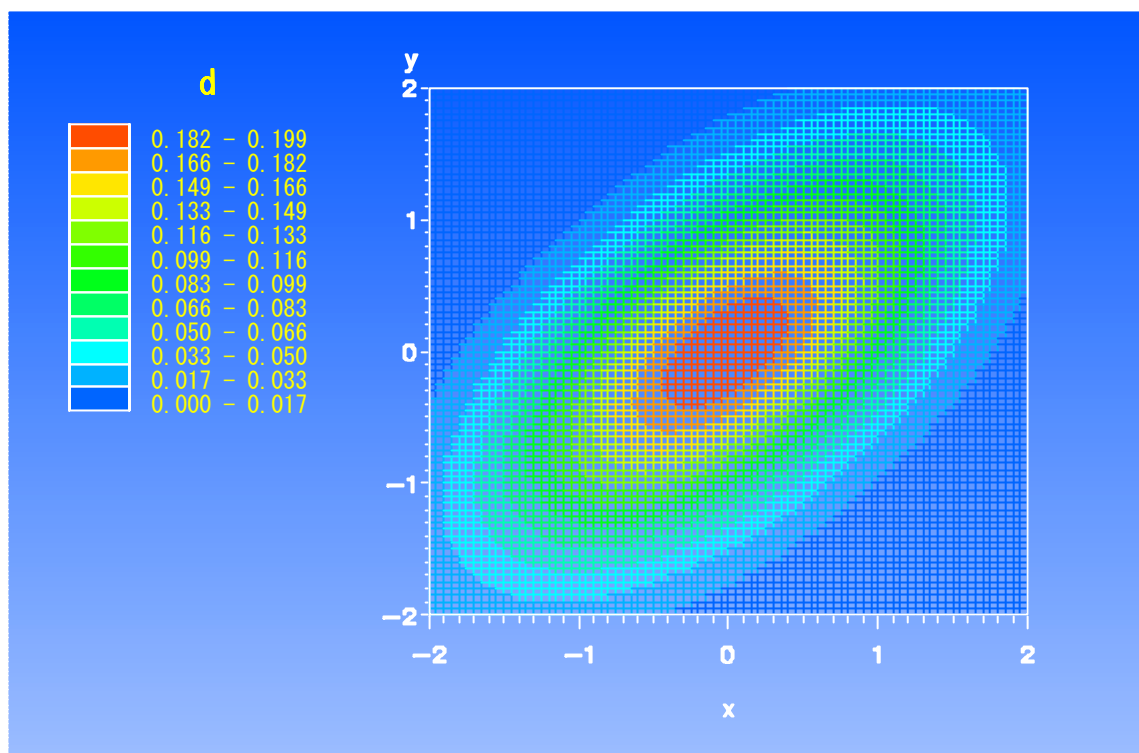


図 3-3 等高線プロット

上記プログラムで作成したデータセットを GXPLOT プロシジャ内にそのまま呼び込めば、簡単に等高線プロットが作成出来てしまうので便利である(図 3-3)。

```

proc gplot data=plotdata;
  plot y*x / annotate=annoall;
run; quit;

```

## 6. まとめ

色彩には定性的要素と定量的要素をそれぞれ自体に秘めており、カラーコードとデータ値との連動により、より恣意的にとらわれにくいグラフ作りができると考えられる。色にはグラフそのものを高品位化したり、形あるものであれば輪郭を強調し、また注意喚起能力も持っている。しかしカラーコードそのものを覚えたとしても SAS の G グラフをそのまま高品位化するのは難しく、外挿用のデータ作りを行わねばならない等の手間は生じる。

今後はカラーコードと ODS 機能や SG グラフとの関連性を調査し、更なるグラフの高品位化を検討したい。

### 参考文献

- ・千々岩英彰 (2001). 色彩学概説, 東京大学出版会
- ・南雲治嘉 (2008). 色の新しい捉え方 現場で「使える」色彩論, 光文社新書
- ・平林雅英 (1992). C 言語による最新プログラム事典 第2巻, 技術評論社
- ・竹内啓監修 (1994). SAS によるデータ解析入門[第2版], 東京大学出版会