

07-1 説明や説得に有効です。

そこで今回は「直線関係の方向と強さ」の指標である相関係数、及び横軸変数から縦軸変数を最適に予測する回帰式の算出機能を前回の script に追加します。

07-2 右の script は・・・

末尾の8行以外はその機能に関して06：のそれと同等です。

07-3 係数や式の算出には・・・

定義式や計算式を得てそれを script にすることが必要です。また正しい使用には概念の意味と適用範囲の理解が不可欠です。

07-4 (積率) 相関係数の定義式は・・・

「**両変数の共分散 ÷ (縦軸変数の SD × 横軸変数の SD)**」です。Script では共分散を $(y-my) \cdot (x-mx)$ の総和で、SD を $(y-my)^2$ 並びに $(x-mx)^2$ の総和の平方根で求め、定義式どおりに算出しています。

共分散は全件が右上がりの対角線※上にある時最大、右下がりの対角線※上にある時最小、それを縦横のSDの積で割るのは値の範囲を -1~+1 にするためです。

今回の素 data 例は中程度の負の相関 (-.60) を示します。
※従って「非直線」の場合には不適合。

その時は08：の相関比を使用。

07-5 回帰式とは・・・

横軸変数から縦軸変数を最も正確 (=ハズレが最も少なくなるよう) に予測する式です※。

※直線 $(y=ax+b)$ を当てはめるなら「一次回帰式」。

計算式は a(傾き) が「**r (相関係数) × 縦変数の SD / 横変数の SD**」、b(切片) が「**縦変数の平均値 - 傾き × 横変数の平均値**」で、今回の素 data 例から得られる

「賭事頻度から幸福度を予測する式」は幸福度 = $-.005 \times \text{賭事頻度} + 6.356$ でした #。

妥当な結果を得るには「多数の“個人 data”」が必要です。

関連情報：

<http://www.iser.osaka-u.ac.jp/rcbe/gyoseki/fukou.pdf>

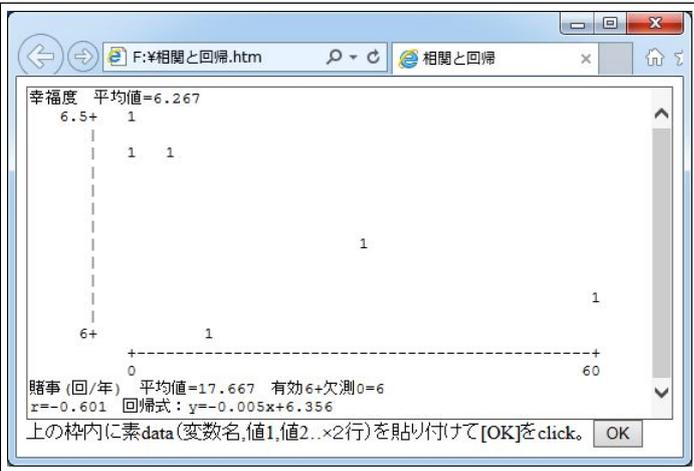
```
<title>相関と回帰</title><!- saved from url=(0008)about:internet ->
<form>
  <textarea rows="18" cols="64"></textarea><br>
  上の枠内に素 data (変数名, 値1, 値2... × 2行) を貼り付けて[OK]をクリック。
  <input type="button" value="OK" onClick="集計()">
</form>

<script>
  素="幸福度, 6.5, 6.4, 6.4, 6.0, 6.2, 6.1 \n 賭事(回/年), 0, 1, 5, 10, 30, 60"
  document.forms[0].elements[0].value=素
  function 集計() {
    素=document.forms[0].elements[0].value.split("\n");初=1;欠=0;有=0
    素[0]=素[0].split(",");縦軸=素[0].shift();縦和=0
    素[1]=素[1].split(",");横軸=素[1].shift();横和=0
    for (i in 素[0]) {if (""=素[0][i]||""=素[1][i]) {欠++;continue}
      if (1=初) {縦大=素[0][i]-0;縦小=縦大;横大=素[1][i]-0;横小=横大;初=0}
      値=素[0][i]-0;縦和=縦和+値; if (値<縦小) 縦小=値; else if (縦大<値) 縦大=値
      値=素[1][i]-0;横和=横和+値; if (値<横小) 横小=値; else if (横大<値) 横大=値
      有++}
    縦単=(縦大-縦小)/12;横単=(横大-横小)/12
    数=new Array();for (i=0;i<13;i++) {数[i]=new Array()}
    for (i=0;i<13;i++) {for (j=0;j<13;j++) {数[i][j]=0}}
    for (i in 素[0]) {if (""=素[0][i]||""=素[1][i]) continue
      y=素[0][i]-0;uy=Math.round((y-縦小)/縦単);
      x=素[1][i]-0;ux=Math.round((x-横小)/横単); 数[uy][ux]++}
    縦平=Math.round(1000*縦和/有)/1000;横平=Math.round(1000*横和/有)/1000
    白=new Array("","","","","","","","","","","")
    線="-----";出=縦軸+" 平均値="+縦平+" \n"
    for (y=12;-1<y;y-) {
      for (x=0;x<13;x++) {区=白[4]
        if (数[y][x]>0) 区=白[4-" "+数[y][x].length]+数[y][x]
        if (y=12 && x=0) 区=白[6-" "+縦大].length]+縦大+" "+区
        if (0<y && y<12 && x=0) 区=白[6+" "+区
        if (y=0 && x=0) 区=白[6-" "+縦小].length]+縦小+" "+区
        出=出+" \n"
        出=出+白[7]+" "+線+線+" "+ \n"
        出=出+白[5]+白[6-" "+横小].length]+横小+白[7]+白[7]+白[7]+白[7]+白[2]
        出=出+白[6-" "+横大].length]+横大+" \n "+横軸+" 平均値="+横平+" 有効"+有
        共=0;分x=0;分y=0;my=縦和/有;mx=横和/有
        for (i in 素[0]) {if (""=素[0][i]||""=素[1][i]) continue
          y=素[0][i];x=素[1][i];共=共+(y-my)*(x-mx)
          分x=分x+(x-mx)*(x-mx);分y=分y+(y-my)*(y-my)}
          r=共/(Math.sqrt(分x)*Math.sqrt(分y));傾き=r*Math.sqrt(分y)/Math.sqrt(分x)
          切片=Math.round(1000*(my-傾き*mx))/1000; if (0<=切片) 切片="+切片
          出=出+" 欠測"+欠+" "+(有+欠)+" \nr="+Math.round(1000*r)/1000+" 回帰式: y="
          document.forms[0].elements[0].value=出+Math.round(1000*傾き)/1000+"x"+切片
        }
      }
    }
  }
</script>
```

共分散/(SDx*SDy)。分子・分母共通の“÷N”は約分により省略。

正確を期して“丸め”は計算終了後に。

資料 17 散布図に加えて (積率) 相関係数と (一次) 回帰式を求める JavaScript の例



資料 18 資料 17 の内容の実行結果

⑮ 数値処理の実行には定義式や計算式が必要です。
⑯ 結果の正しい利用には意味の理解が不可欠です。