

Tikz サンプル

鷹合孝之

2018 年 12 月 5 日

1 はじめに

1.1 各図形の書式についての簡単な説明

1. 方眼の書式

- 方眼は `\draw (左下隅) grid (右上隅);`
 - オプションの `[help lines]` によって、方眼がグレーの細線になる
 - さらに `[step = 5mm]` などとして方眼の間隔を設定できる（デフォルトでは 1cm）

2. 直線の書式

- 線分は `\draw (始点) – (中間点) … – (終点);`
 - 点と点のあいだの `–` は「ハイフン 2 個」
 - パスを閉じるときは最後に `–cycle` をつける（このオプションで始点に戻ることになる）
 - 滑らかな角にするにはその手前で `[rounded corners] –` とする
 - 再び直線的な角にするにはその手前で `[sharp corners] –` とする
 - (終点) の座標を極座標形式 (角度:長さ) で表現することもでき、`++` をつければ始点からの相対座標となる（なければ絶対座標）
- 長方形は `\draw (左下隅) rectangle (右上隅);`
- タテ・ヨコだけで結ぶ線は `\draw (始点) — (終点);` または `\draw (始点) — (終点);`
 - `”–”` が (始点) と (終点) をまっすぐに結ぶのに対し、`—` は「始めにタテ、次にヨコ」で 2 点を結び、`–` は「始めにヨコ、次にタテ」で 2 点を結ぶ

3. 円、楕円、弧（扇形）の書式

- 円は `\draw (中心) circle [radius=r];`
 - 古い仕様では、半径を (r) とつけることになっていたが、可読性の観点から座標以外を () で囲むことは現在は推奨されていない
 - 座標上の「点」を黒丸で示すときには旧形式の表記がよく用いられる (`\fill (中心) circle (2pt);`)

- 楕円は `\draw (中心) circle [x radius = x 方向の径, y radius = y 方向の径, rotate = 回転角];`
 - － 回転角はラジアン (弧度法) ではなく, 度数法 (1 回転が 360 度となるもの) で表す
- 弧は `\draw (始点) arc [start angle = 始点の角度, end angle = 終点の角度, radius = 半径];`
 - － 角度はラジアン (弧度法) ではなく, 度数法 (1 回転が 360 度となるもの) で表す
 - － 略式では `\draw (始点) arc (始点の角度:終点の角度:半径);` とすることもできる
 - － 始点の角度または終点の角度の一方の代わりに `delta angle = 中心角` を設定することもできる
 - － `radius` の代わりに `x radius` と `y radius` に異なった長さを設定すれば楕円の弧の一部になる

4. 放物線, サインカーブ, 曲線, ベジエ曲線の書式

- 放物線は `\draw (始点) parabola bend (頂点) (終点);`
 - － ただし, 上の式は「始点から頂点までの放物線」と「頂点から終点までの放物線」を表し, 必ずしも 3 点を通る 1 つの放物線になるとは限らない (同一放物線にするには, 適切な座標を記述者自身が与えなくてはならない)
 - － `\draw (始点) parabola (終点);` と 2 点のみを与えた場合は始点が頂点, `\draw (始点) parabola [bend at end] (終点)` とした場合は終点が頂点になる
 - － `\draw (始点) parabola [parabola height = 高さ] +(幅,0);`
- サインカーブは `\draw (始点) sin (頂点);` または `\draw (頂点) cos (終点);`
 - － `sin` を用いれば「原点から頂点まで」の $1/4$ 周期を描き, `cos` を用いれば「頂点から原点まで」の $1/4$ 周期を描く (いずれも $[0, \pi/2]$ の範囲ということ)
 - － 三角関数の 1 周期分を記述するには, `sin` と `cos` を適切に組み合わせる
 - － `sin` および `cos` で描かれるのは三角関数の $1/4$ 周期分のみ
- 曲線として `\draw (始点) to [out = 始点から出る角度, in = 終点に入る角度] (終点);`
 - － 2 点の「出」と「入り」の角度を指定し, 滑らかな曲線でつなぐ
 - － `[out = ○, in = ○]` というオプションがなければ, 上式は `\draw (始点) -- (終点);` に等しい
- `[bend 方向 (= 角度) , distance = 距離]` というオプションで 2 点間を緩やかに膨らんだ曲線で結ぶ
 - － 方向は (進行方向に対して) `left` または `right`
 - － 角度を指定すると `[out = 角度, in = 180 - 角度]` を意味する
 - － 角度を省略すると, 以前の数値が機械的に使われる
 - － `distance` で膨らみ具合を調整できる (ベジエ曲線の `controls` の距離を調整している)
- ベジエ曲線は `\draw (始点) .. controls (方向点 1) and (方向点 2) .. (終点);`
 - － 3 次のベジエ曲線になる

- 方向点 2 を省略すると、方向点 2 と終点が一致していると見なされて描画される

5. 関数による描画

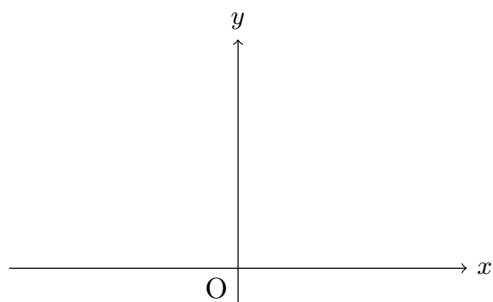
- 座標指定の y 座標が x 座標 ($\backslash x$) の関数になるように記述する
 - デフォルトの変数は $\backslash x$ だが、オプションで `variable = \t` などとすれば変えられる
 - 変域のデフォルトは $[-5:5]$ だが、オプションで `domain = jstart:jend` と指定できる
 - 必要に応じて中括弧 () を使用する
- ここでは PGF (TikZ) の計算機能を利用している (gnuplot などの外部エンジンの利用も可能)
 - 使用可能な PGF の計算機能についてはマニュアルの第 89 章 Mathematical Expressions を参照

2 例

2.1 座標軸

LaTeX

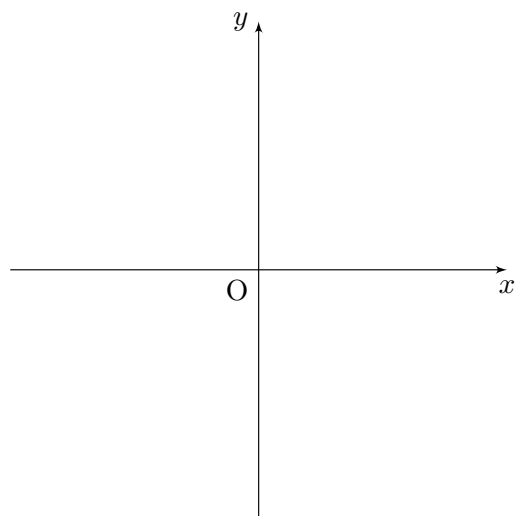
```
\begin{tikzpicture}
\datavisualization[
school book axes,
x axis={label={$x$}, ticks={major={at={}}}},
y axis={label={$y$}, ticks={major={at={}}}},
]
data[
x,y
-2.5, -2.5
2.5, 2.5
];
\begin{scope}
\draw (0,0) node[below left]{0}; % 原点
\end{scope}
\end{tikzpicture}
```



LaTeX

```
\begin{tikzpicture}[> = latex']
\node(0) at (0,0) [below left]{$\mathrm{0}$};
\draw[->] (-3,0) -- (3,0) node[below]{$x$};
\draw[->] (0,-3) -- (0,3) node[left]{$y$};

\end{tikzpicture}
```



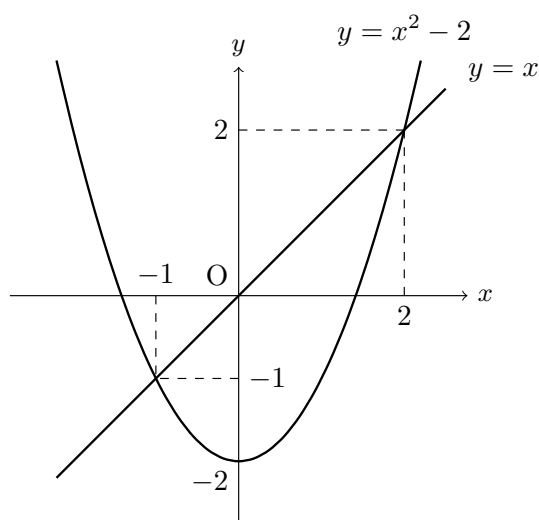
2.2 曲線

LaTeX

```
\begin{tikzpicture}
\datavisualization[
school book axes,
x axis={label={\$x\$},ticks={major={at={}}}},
y axis={label={\$y\$},ticks={major={at={}}}},
]
data{
x,y
- 2.5, - 2.5
2.5, 2.5
};
\begin{scope}
\draw (0,0) node[above left]{0};

\draw[thick, domain=-2.2:2.5] plot (\x,\x) node at(3.2,2.7) {\$y = x\$};
\draw[thick, domain=-2.2:2.2,smooth] plot (\x,{\x * \x - 2}) node at(2,3.2) {\$y = x^2 - 2\$};

\draw[thin,dashed] (2,0) node [below]{\$2\$}--(2,2);
\draw[thin,dashed] (0,2) node [left]{\$2\$}--(2,2);
\draw[thin,dashed] (-1,0) node [above]{\$ -1\$}--(-1, -1);
\draw[thin,dashed] (0, -1) node [right]{\$ -1\$}--(-1, -1);
\draw (0, -2) node [below left]{\$ -2\$};
\end{scope}
\end{tikzpicture}
```



LaTeX

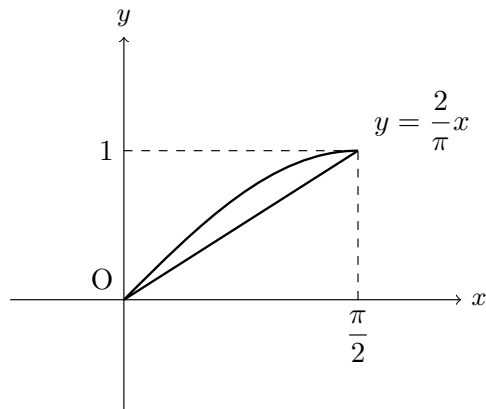
```
\begin{tikzpicture}[scale = 1.8]
\datavisualization[
school book axes,
x axis={label={\x$},ticks={major={at={}}}},
y axis={label={\y$},ticks={major={at={}}}},
]
data{
x,y
- .5, - .5
2, 1.5
};
\begin{scope}
\draw (0,0) node[above left]{0};

\draw[thick, domain=-0:pi/2] plot (\x,{(2/pi) * \x}) node at(2,1.2) {\$y = \dfrac{2}{\pi}x\$};
% \draw[thick, domain=-2.2:2.2,smooth] plot (\x,{\x * \x - 2}) node at(2,3.2) {\$y = x^2 - 2\$};

\draw[thick] (0, 0) sin (pi/2,1);

\draw[thin,dashed] (pi/2,0) node [below]{\dfrac{\pi}{2}}--(pi/2,1);
\draw[thin,dashed] (0,1) node [left]{1}--(pi/2,1);

% \draw (0, -2) node [below left]{\$ -2\$};
\end{scope}
\end{tikzpicture}
```



2.3 領域

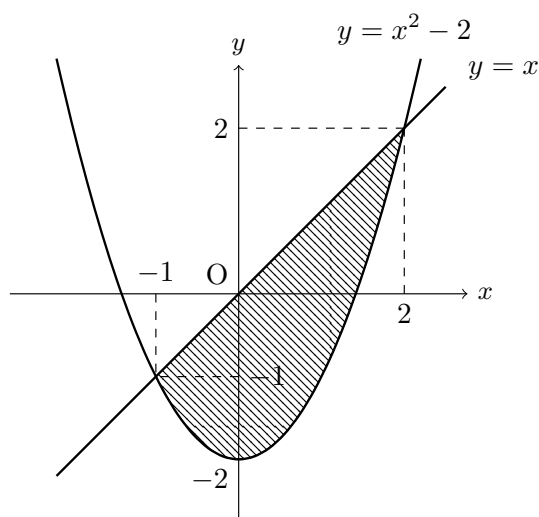
LaTeX

```
\begin{tikzpicture}
\datavisualization[
school book axes,
x axis={label={\x$},ticks={major={at={}}}},
y axis={label={\y$},ticks={major={at={}}}},
]
data{
x,y
- 2.5, - 2.5
2.5, 2.5
};
\begin{scope}
\draw (0,0) node[above left]{0};

\draw[thick, domain=-2.2:2.5] plot (\x,\x) node at(3.2,2.7) {\$y = x\$};
\draw[thick, domain=-2.2:2.2,smooth] plot (\x,{\x * \x - 2}) node at(2,3.2) {\$y = x^2 - 2\$};

\draw[thin,dashed] (2,0) node [below]{\$2\$}--(2,2);
\draw[thin,dashed] (0,2) node [left]{\$2\$}--(2,2);
\draw[thin,dashed] (-1,0) node [above]{\$ -1\$}--(-1, -1);
\draw[thin,dashed] (0, -1) node [right]{\$ -1\$}--(-1, -1);
\draw (0, -2) node [below left]{\$ -2\$};

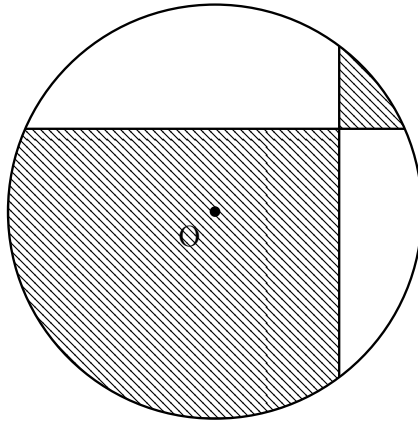
\clip (-2.5,-2.5) |- (0, - 2.5) |- (2.5,2.5) --cycle;
\clip [draw,domain= -2:2.2] plot (\x,{\x * \x - 2});
\draw [pattern=north west lines](-2.5, -2.5) rectangle (2.5,2.5);
\end{scope}
\end{tikzpicture}
```



LaTeX

```
\begin{tikzpicture}[scale=.5]
\path[clip, preaction={draw, thick}] (0,0) circle (5);
\fill[draw=black, thick, pattern=north west lines] (-5,2) -- (3,2) -- (3,-5) -- (-5,-5) -- cycle;

\fill[draw=black, thick, pattern=north west lines] (3,5) -- (3,2) -- (5,2) -- (5,5) -- cycle;
\node[draw, circle, thick, fill=black, minimum size=1mm, inner sep=0, label=225:O] at (0,0) {};
\end{tikzpicture}
```



2.4 図形

