This document is a collection of the tikz code I’ve found useful while writing lecture notes and exams. Of course, the definitive reference is the Tikz & PGF Manual. The most recent version of that manual I am aware of is version 2.10.

Contents

1 Basic Stuff ......................................................... 2
  1.1 grid .......................................................... 2
  1.2 axes with tick marks ......................................... 2
  1.3 lines and arrows .............................................. 2
  1.4 braces ....................................................... 2
  1.5 coordinates and nodes ....................................... 3
  1.6 perpendicular lines .......................................... 4

2 Circles and polar coordinates .................................. 4
  2.1 arrows along a circle ....................................... 4
  2.2 polar coordinates ........................................... 4
  2.3 arcs ......................................................... 5

3 Curves .......................................................... 6
  3.1 parabola ..................................................... 6
  3.2 polynomial-like curves ..................................... 7
    3.2.1 smooth curve through a list of points ............... 7
    3.2.2 using controls ......................................... 7
    3.2.3 entrance and exit angles through points .......... 9
    3.2.4 graph a function from the equation ............... 9
    3.2.5 example: polynomial .................................. 10
  3.3 sine and cosine curves .................................... 11
  3.4 putting a coordinate along a curve ..................... 11
  3.5 intersection points of two curves ....................... 12

4 Random Stuff .................................................... 13
  4.1 fill an area ................................................ 13
  4.2 tangent lines .............................................. 14
  4.3 cone ....................................................... 16
  4.4 3D pictures ............................................... 16

5 Examples ........................................................ 17
  5.1 Example: Linear Approximation ........................... 18
  5.2 Example: Mean Value Theorem ............................. 19
1 Basic Stuff

1.1 grid

\begin{tikzpicture}[scale=.5]
  \draw[dashed, gray] (0,0) grid (7,7);
\end{tikzpicture}

1.2 axes with tick marks

\begin{tikzpicture}[scale=.7]
  \draw[->] (0,-4) -- (0,4);
  \draw[->] (-4,0) -- (4,0);
  \foreach \x in {-3,-2,-1,1,2,3}
    \draw (\x,2pt) -- (\x,-2pt);
  \foreach \y in {-3,-2,-1,1,2,3}
    \draw (2pt,\y) -- (-2pt,\y);
\end{tikzpicture}

1.3 lines and arrows

\begin{tikzpicture}
  \draw[<->] (0,0) -- (2,0);
  \draw[|-<->|] (0,0) -- (2,0);
  \draw[thick] (0,0) -- (2,0);
  \draw[dashed] (0,0) -- (2,0);
  \draw (0,0) -- (2,0);
  \draw[shorten <= .25cm, shorten >= .5cm] (0,0) -- (2,0);
  \draw[shorten <= -.5cm, shorten >= -1cm] (0,0) -- (2,0);
\end{tikzpicture}

To use some of these, you need to include the library \usetikzlibrary{arrows}.

\begin{tikzpicture}
  \draw[->, >= angle 90] (0,0) -- (2,0);
  \draw[->, >= triangle 90] (0,0) -- (2,0);
\end{tikzpicture}

There's a nice list of arrow types from the arrows library here.

1.4 braces

To add braces to a tikz picture, you must include the package \usetikzlibrary{snakes}
Here is a simple example:
\draw[thick, decoration={brace, mirror}, decorate] (0,0) -- (3,0);
\begin{tikzpicture}
  \draw[thick, decoration={brace, mirror}, decorate] (0,0) -- (3,0);
\end{tikzpicture}

Taking away 'mirror' gives us:
\draw[thick, decoration={brace}, decorate] (0,0) -- (3,0);
\begin{tikzpicture}
  \draw[thick, decoration={brace}, decorate] (0,0) -- (3,0);
\end{tikzpicture}

1.5 coordinates and nodes

Nodes and coordinates. I generally use node to label locations on a graph.

\node (name) at (coordinate) {caption};
\coordinate (name) at (coordinate);
\coordinate (A) at (1,1);
\coordinate (B) at (3,2);
\filldraw (A) circle (2pt) node[right] {A};
\filldraw (B) circle (2pt) node[right] {B};
\begin{tikzpicture}
  \coordinate (A) at (1,1);
  \coordinate (B) at (3,2);
  \filldraw (A) circle (2pt) node[right] {A};
  \filldraw (B) circle (2pt) node[right] {B};
\end{tikzpicture}

You can name a coordinate within a node:
\draw[->] (-.5,0) -- (3,0) node[right](xline) {x};
\draw[->] (0,-.5) -- (0,3) node[right](yline) {y};
\draw[red] (xline) -- (yline);
\begin{tikzpicture}
  \draw[->] (-.5,0) -- (3,0) node[right](xline) {x};
  \draw[->] (0,-.5) -- (0,3) node[right](yline) {y};
  \draw[red] (xline) -- (yline);
\end{tikzpicture}

I can make some of these things some colors.
\coordinate (A) at (1,1);
\coordinate (B) at (3,2);
\filldraw (A) circle (2pt) node[right, blue] {A};
\filldraw[red] (B) circle (2pt) node[right] {B};
\begin{tikzpicture}
  \coordinate (A) at (1,1);
  \coordinate (B) at (3,2);
  \filldraw (A) circle (2pt) node[right, blue] {A};
  \filldraw[red] (B) circle (2pt) node[right] {B};
\end{tikzpicture}
1.6 perpendicular lines

\node[above] at (A |- B) {A \vline \ - B};
\node[below] at (A -| B) {A \ - \vline\ B};
\draw[dashed] (B) -- (B |- A);

2 Circles and polar coordinates

2.1 arrows along a circle
I can draw an arrow at a point along a circle:

\draw[
    decoration={markings, mark=at position 0.2 with {\arrow[line width=1mm]{>}}},
    postaction={decorate},
    thick
] (0,0) circle (1);

\draw[
    decoration={markings, mark=at position 0.825 with {\arrowreversed[line width=1mm]{>}}},
    postaction={decorate},
    thick
] (0,0) circle (1);

2.2 polar coordinates
We can also specify coordinates in (angle : radius) form, with the angle given in degrees.
2.3 arcs

Draw an arc with \texttt{\textbackslash draw (coordinate) arc (start angle : end angle : radius)}. 
3 Curves

3.1 parabola

\draw[thick] (-1,-2) parabola bend (1,2) (3,-2);
Draw a parabola:

(vertx) parabola (point)
\draw[thick] (0,3) parabola (2,0);
\draw[thick] (4,-3) parabola (2,0);
\draw[thick] (4,-3) parabola (6,0);
\draw[thick] (7,2) parabola (6,0);

\begin{tikzpicture}
\end{tikzpicture}

3.2 polynomial-like curves

3.2.1 smooth curve through a list of points
\draw[thick] plot[smooth, tension=.7] coordinates {(-1.5,3) (0,-1) (3,2.5) (5,-3)};
\begin{tikzpicture}
\end{tikzpicture}

\draw[thick] plot[smooth] coordinates {(0,2) (1,3) (2,1.5) (3,2) (5,-1) (6,0)};
\begin{tikzpicture}
\end{tikzpicture}

3.2.2 using controls
\draw (0,0) .. controls (1,1.5) .. (3,2);
\draw (0,0) .. controls (-1,1.5) .. (-3,2);
\draw (0,0) .. controls (1.5,.5) .. (3,3);

Here's an example using a parabola and controls:
\begin{tikzpicture}[scale=.7]
\draw[->] (-.5,0) -- (12.5,0) node[right] {$t$ (sec)};
\draw[->] (0,-4.5) -- (0,8.5) node[above] {$v(t)$ (ft/s)};
\draw[dashed, thin] (0,-4) grid (12,8);
\foreach \x in {1,2,3,4,5,6,7,8,9,10,11,12}
\draw (\x,2pt) -- (\x, -2pt) node[below] {\x};
\foreach \y in {-100,-50,50,100,150,200}
\draw (2pt,\y/25) -- (-2pt, \y/25) node[left] {\y};
\draw[thick] (0,0) parabola (2,7);
\draw[thick, shorten >= -3.2cm] (2,7) -- (8,0);
\draw[thick] (11,-3.5) .. controls (11.5,-2) .. (12,-1.5);
\end{tikzpicture}
3.2.3 entrance and exit angles through points

\draw[thick] (-3,-2) to (-.7,2.5)
    to[out=60, in=180] (0,3)
    to[out=0, in=170] (2,2.2)
    to[out=-10, in=0] (5,2.1);
\draw[dashed] (-4,2) -- (5,2);

3.2.4 graph a function from the equation

\draw plot[domain=-.5:3.5] (\x, {.5*exp(.5*\x)}) node[right] {$y=f(t)$};

\draw[thick] plot[samples=100, smooth, domain=-3:3.8] (\x, {.05*(\x+2.6)*(-\x-3.7)*\x^3});
There can be weird issues with this. For example, graphing
\[ y = \frac{2}{1 + x^2} \]
with
\begin{verbatim}
\draw[thick] plot[samples=100, smooth, domain=-4:4] (x, {2/(1+x^2)});
\end{verbatim}
gives a weird giant spike at \( x = 0 \). The fix is to instead graph:
\begin{verbatim}
\draw[thick] plot[samples=100, smooth, domain=.02:4] (x, {2/(1+x^2)});
\draw[thick] plot[samples=100, smooth, domain=.02:4] (-x, {2/(1+x^2)});
\end{verbatim}

\subsection{3.2.5 example: polynomial}
\begin{verbatim}
\begin{tikzpicture}[xscale=.7, yscale=.5]
\draw[->] (-5,0) -- (6,0) node[right] {$x$};
\draw[->] (0,-5) -- (0,5) node[right] {$y$};
\draw[thin, dashed, gray] (-5,-5) grid (6,5);
\draw[thick] plot[samples=100, smooth, domain=-4.47:6] (x, {.02*(x+3)*(x-1)*(x-6)*(x+1.5)});
\foreach x in {-5,-4,...,6}
\draw (\x,2pt) -- (\x,-2pt) node[below] {$x$};
\end{tikzpicture}
\end{verbatim}
3.3 sine and cosine curves

\begin{center}
\begin{tabular}{|l|c|}
\hline
\verb|\draw (0,0) sin (1,1);| & \verb|\draw (0,1) sin (1,0);| \\
\hline
\verb|\draw (0,0) cos (1,1);| & \verb|\draw (0,1) cos (1,0);| \\
\hline
\end{tabular}
\end{center}

We can put a bunch of these together to draw a \textit{sin} or \textit{cos} curve.

\verb|\draw (0,0) sin (1,1) cos (2,0) sin (3,-1) cos (4,0);|
\verb|\draw (0,0) sin (-1,-1) cos (-2,0) sin (-3,1) cos (-4,0);|

3.4 putting a coordinate along a curve

When drawing a curve, you can put a coordinate at some point along the curve. For instance, \verb|coordinate[pos=.2] (A)| puts a coordinate \( \frac{1}{5} \) of the way along the curve. Here are some examples:

\verb|\draw (-.5,1.2) to[out=0, in=180] coordinate[pos=.5] (A) (5,2.8);|
\verb|\draw[dashed] (A) -- (A |- xline) node[below] {$a$};|
I can put a bunch of them all on the same curve:

\draw (-3,1) to[out=0, in=180]
coordinate[pos=0] (A)
coordinate[pos=.17] (B)
coordinate[pos=.5] (C)
coordinate[pos=.67] (D)
coordinate[pos=1] (E)
(3,-1);

3.5 intersection points of two curves

Use \path[name intersections={of=CURVE1 and CURVE2}] to find coordinates of the intersections of two
curves.

\draw[name path=curve, samples=50, smooth, domain=0:6] plot (\x, {-.1*(\x)*((\x-3)*((\x-6)+2)});

\path[name path=line1] (1,0) -- ++(0,6);
\path[name intersections={of=curve and line1}];
\filldraw (intersection-1) circle (2pt) node[below] {intersection-1};

\path[name path=line5] (5,0) -- ++(0,6);
\path[name intersections={of=curve and line5}];
\filldraw (intersection-1) circle (2pt) node[above] {intersection-1};
4 Random Stuff

4.1 fill an area

You can use \texttt{\texttt{\texttt{fill}}} to fill in an enclosed area:

\begin{verbatim}
\fill[fill=gray] (0,0) -- (0,2) -- (3,2) -- (3,0) -- cycle;
\end{verbatim}

\begin{tikzpicture}[yscale=3, xscale=4]
\fill[fill=gray!50] (0,0) -- (0,1) -- plot [domain=0:1] (x,\x^2+1) -- (1,0) -- cycle;
\end{tikzpicture}

I once tried to fill the space under a curve with vertical lines. Instead of using \texttt{\texttt{\texttt{fill}}} with something, I just drew a bunch of vertical lines using \texttt{name intersections} as described in section 3.5.
4.2 tangent lines

Drawing a tangent line to a graph. I think there must be a standard, easier way to draw a tangent line. I’d appreciate an email if you know how to do it! The solution here was found on the following stackexchange question: link.

First, you add the following option to the tikzpicture:

\begin{tikzpicture}
\tangent/.style={
\t    decoration={
\t\tmarkings,\%
\t\tswitch on markings
\t\tmark=
\t\t\at position \#1
\t\t\with
\t\t\{
\t\t\coordinate (tangent point-\pgfkeysvalueof{/pgf/decoration/mark info/sequence number}) at (0pt,0pt);
\t\t\coordinate (tangent unit vector-\pgfkeysvalueof{/pgf/decoration/mark info/sequence number}) at (1,0pt);
\t\t\coordinate (tangent orthogonal unit vector-\pgfkeysvalueof{/pgf/decoration/mark info/sequence number}) at (0pt,1);
\t\t\}
\t\},
\t\postaction=decorate
\t},
\t\use tangent/.style={
\t\t\shift=(tangent point-#1),
\t\tx=(tangent unit vector-#1),
\t\ty=(tangent orthogonal unit vector-#1)
\t\},
\t\use tangent/.default=1
\t}\end{tikzpicture}
First you say where the tangent line is along the curve by adding `tangent=POSITION` as an option to the \draw command, where POSITION is the fraction of the curve before the tangent line. For example, \draw[tangent=.7] if you want to draw a tangent line 7/10 of the way along the curve.

Then, when you want to draw the tangent line, use the option `use tangent` on the draw command. This will draw according to a new coordinate system, with the tangent point you defined earlier at (0, 0).

\draw[tangent=0.4] (1,1) to[out=70, in=200] (4,4);
\filldraw[use tangent] (0,0) circle (2pt);
\draw[use tangent] (-2,0) -- (2.5,0);

\draw[tangent=0.4] (.5,2) to[out=-60,in=170] (1.5,1) to[out=10,in=-120] (2.5,2);
\filldraw[use tangent] (0,0) circle (2pt);
\draw[use tangent] (-1,0) -- (1,0);

\draw[tangent=0.0] (0,0) sin (1,1) cos (2,0);
\draw (0,0) sin (-1,-1) cos (-2,0);
\draw[use tangent] (-2,0) -- (2,0);
4.3 cone

First draw the bottom of the cone with an arc, and the sides with some lines:
\draw (-1,0) arc (180:360:1cm and 0.5cm) -- (0,2) -- cycle;

Draw the back of the cone with another arc.
\draw (-1,0) arc (180:360:1cm and 0.5cm) -- (0,2) -- cycle;
\draw[dashed] (-1,0) arc (180:0:1cm and 0.5cm);

You get:

4.4 3D pictures

I haven’t quite been able to figure out the options at the beginning of these 3D plots. But here are two simple examples.

\begin{tikzpicture}[cm={-1,-1,1,0,(0,0)},x=3.85mm,z=-1cm]
    \draw[thick,->,black] (0,0,0) -- (4,0,0) node[anchor=north east]{north};
    \draw[thick,->] (0,0,0) -- (0,3,0) node[anchor=north east]{west};
    \draw[thick,->] (0,0,0) -- (0,0,3) node[anchor=east]{up};
    \draw (0,2,0) -- node[anchor=north west] {$\tilde{D}$} (3,0,0)
      -- node[left] {4} (3,0,1.5)
      -- node[near start, above] {$D$} (0,2,0);
    \node[above] at (0,1,0) {$\delta_T$};
    \node[left] at (1,0,0) {$\delta_P$};
\end{tikzpicture}
5 Examples

Here are some random examples that put together a bunch of stuff.
5.1 Example: Linear Approximation

The tikz code is the following, plus that tangent line option I mentioned at the beginning of section 4.2.

\draw[->] (-.5,0) -- (6,0) node(xline)[right] {$x$};
\draw[->] (0,-.5) -- (0,6) node(yline)[right] {$y$};

\draw[tangent=0.2] (1,1) to[out=-30,in=270]
  coordinate[pos=0.2] (A)
  coordinate[pos=0.7] (B)
  (5,5) node[right] {$y=f(x)$};

\draw[use tangent, name path=tan] (-2,0) -- (3.5,0) node[right] {$y=L(x)$};
\draw[dashed] (A) -- (A |- xline) node[below] {$a$};
\draw[dashed, name path=horiz, shorten <= -.2cm]
  (A |- yline) node[left, xshift=-.2cm] {$f(a)$} -- (A -| B);
\draw[dashed, name path=vert] (B) -- (B |- xline) node[below] {$x$};
\fill[name intersections={of=horiz and vert}] (intersection-1) circle (.2pt) coordinate(D);
\fill[name intersections={of=tan and vert}] (intersection-1) circle (.2pt) coordinate(C);
\draw[thick, decoration={brace, mirror, raise=.2cm}, decorate]
  (E) -- node[below, yshift=-.3cm] {$x-a$} (F);
\draw[thick, decoration={brace, raise=.2cm}, decorate]
  (B) -- node[right, xshift=.2cm] {$E(x)$} (C);
5.2 Example: Mean Value Theorem

\[ y = f(x) \]

\[ \text{\textbackslash{draw}}[-.5,0] -- (7,0) \text{node(xline)[right]} \{\$x\$}; \]
\[ \text{\textbackslash{draw}}[-.5] (0,-.5) -- (0,5) \text{node(yline)[right]} \{\$y\$}; \]

\[ \text{\textbackslash{draw}[thick, blue, smooth, tangent=0.255, tangent=0.745]} (1,2) \]
\[ \text{to[out=-60, in=170] coordinate[pos=0.5]} (A) (2,1) \]
\[ \text{to[out=10, in=190]} (5,4) \]
\[ \text{to[out=-10, in=120] coordinate[pos=0.5]} (B) (6,3) \text{node[right]} \{\$y=f(x)\$}; \]

\[ \text{\textbackslash{fill}} (A) \text{circle (2pt)}; \]
\[ \text{\textbackslash{fill}} (B) \text{circle (2pt)}; \]

\[ \text{\textbackslash{draw}[shorten <= -.5cm, shorten >= -.5cm]} (A) -- (B); \]

\[ \text{\textbackslash{draw}[thick, use tangent]} (-1,0) -- \text{coordinate (C) (1,0)}; \]
\[ \text{\textbackslash{draw}[thick, use tangent=2]} (-1,0) -- \text{coordinate (D) (1,0)}; \]

\[ \text{\textbackslash{fill}} (C) \text{circle (2pt)}; \]
\[ \text{\textbackslash{fill}} (D) \text{circle (2pt)}; \]

\[ \text{\textbackslash{draw}[dashed]} (A) -- (A \text{\textbackslash{xline}}) \text{node[below]} \{\$a\$}; \]
\[ \text{\textbackslash{draw}[dashed]} (B) -- (B \text{\textbackslash{xline}}) \text{node[below]} \{\$b\$}; \]

\[ \text{\textbackslash{draw}[dashed]} (C) -- (C \text{\textbackslash{xline}}) \text{node[below]} \{\$c_1\$}; \]
\[ \text{\textbackslash{draw}[dashed]} (D) -- (D \text{\textbackslash{xline}}) \text{node[below]} \{\$c_2\$}; \]