

[Inv2_Tang_Derivative.mws](#)

- [Find the intervals of increase and decrease](#)
- [Find the inverse functions by solving the equation \$y=f\(x\)\$ for \$x\$ as a function of \$y\$](#)
- [Compute the tangent lines to \$f\$ at selected points in each interval](#)
- [Find the corresponding tangent lines to the inverses. Verify the derivatives predicted by the Inverse Function Theorem.](#)
- [Plot it all](#)
- [Find a point where corresponding tangent lines intersect](#)
- [Prove a simple, yet general, theorem with Maple](#)
- [Generalize inverse selection with RootOf](#)

Calculus II -- worksheet 2

Inverse functions, their tangent lines, and their derivatives

Author: Carl Devore <devore@math.udel.edu>

13 February 2002

Audience:

Second-semester calculus students.

Objectives:

1. Learn how to restrict the domain of a function so that an inverse can be defined.
2. Explore graphical properties of inverse functions.
3. Verify the Inverse Function Theorem.
4. Learn how to put together a complicated plot.
5. Show how Maple can be used to prove a simple, yet general, theorem.

New Maple Commands/Concepts used:

restart, solve, D, diff, evalf, simplify, unapply, is, limit, {}, [], %, #, ||, ', if, map, seq, zip, eval, subs, min, max, with, display, RootOf

The problem (in general):

Take a function and over each interval of increase or decrease define an inverse. Plot the function, the inverses, their derivatives, and some selected tangent lines.

It is a good idea to start most worksheets with the 'restart' command. This clears the values of previously defined variables, which is not automatically done when you open a new worksheet.

> **restart;**

Maple does not provide any response to a restart command.

The function that I want to start with is $\frac{x}{x^2 + 1}$.

Find the intervals of increase and decrease

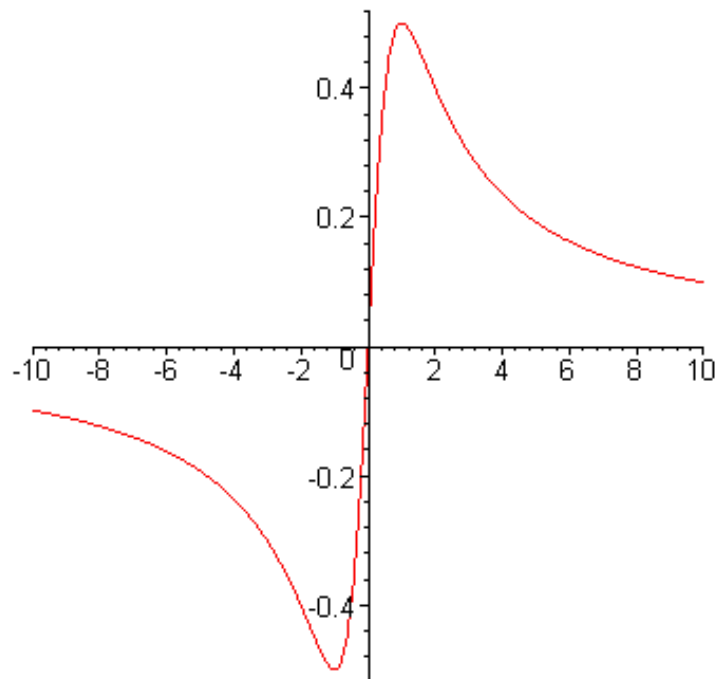
First define the function as a Maple operator, rather than as an expression. (What we ordinarily call functions, Maple calls operators, whereas the word function itself has a slightly different meaning to Maple.)

> **f:= x-> x/(x^2+1);**

$$f := x \rightarrow \frac{x}{x^2 + 1}$$

To get some idea of what we are working with, I'll plot the function over an arbitrary range.

> **plot(f, -10..10);**



So, there are three intervals of increase or decrease. Let's get the critical points. First, I'll get the derivative and call it Df. The command **D** takes the derivative of an operator (whereas the command **diff** takes the derivative of an expression).

> **Df:= D(f);**

$$Df := x \rightarrow \frac{1}{x^2 + 1} - \frac{2x^2}{(x^2 + 1)^2}$$

Clearly, that expression could be and should be simplified. Df is a operator, and the **simplify** command won't work on it. I'll have to work with an expression and then convert it to a function. Note that if f is an operator, then $f(x)$ is an expression.

> **f(x);**

$$\frac{x}{x^2 + 1}$$

And I take the derivative of that with **diff**.

> **Df:= diff(f(x), x);**

$$Df := \frac{1}{x^2 + 1} - \frac{2x^2}{(x^2 + 1)^2}$$

Note that **diff** requires a second argument: The final x tells it to take the derivative with respect to x . Let's simplify it.

> **Df:= simplify(Df);**

$$Df := -\frac{x^2 - 1}{(x^2 + 1)^2}$$

Note that I reused the name Df (because I have no interest in saving the unsimplified form). Let's find the critical points. The second argument to **solve** is the variable to solve for.

> **CritPts:= solve(Df=0, x);**

$$CritPts := -1, 1$$

Now if we combine that information with the plot, it is clear that f is one-to-one (injective) on each of the domains $(-\infty, -1)$, $[-1, 1]$, $(1, \infty)$. (Henceforth, I will use the word "injective" rather than the abomination "one-to-one".)

Furthermore, f is still injective on the two-interval domain $(-\infty, -1) \cup (1, \infty)$.

As I am writing this, -1 is the answer on the left. However, I cannot know if that will be true for you; **solve** does not necessarily give the answers in numerical order. I want a to be the lesser critical point and b the greater. The following command will ensure that.

> **a:= min(CritPts); b:= max(CritPts);**

$$a := -1$$

$$b := 1$$

Note that **min** and **max** just find the smallest and largest values in a finite sequence. This is not the same as finding minima or maxima of a function.

Find the inverse functions by solving the equation $y=f(x)$ for x as a function of y

We use **solve**. If you do the algebra on paper (you should do it!), you will see that you get a quadratic equation in x . Therefore, you should expect **solve** to give two answers. Also, the number of answers should be the maximal number of times that any horizontal line crosses the graph of f , which is also the minimal number of domain restrictions needed to divide f into injective pieces.

> **solve(y=f(x), x);**

$$\frac{1}{2} \frac{1 + \sqrt{1 - 4y^2}}{y}, \frac{1}{2} \frac{1 - \sqrt{1 - 4y^2}}{y}$$

Now I want to manipulate those answers individually. I have to save those answers as a set and give the set a name. Anything can be made into a set by enclosing it in $\{\}$. The result of the last computation is accessed by $\%$. (This is like using the ans key on a TI calculator).

> **inverses := { % };**

$$\text{inverses} := \left\{ \frac{1}{2} \frac{1 + \sqrt{1 - 4y^2}}{y}, \frac{1}{2} \frac{1 - \sqrt{1 - 4y^2}}{y} \right\}$$

Warning: $\%$ refers to the most recent result that you actually computed. That is not necessarily the same thing as the result that appears immediately above your current position on the screen. It is very common for new users to make mistakes because of this.

You can extract the members of a set or a list by using $[\]$:

> **inverses[1];**

$$\frac{1}{2} \frac{1 + \sqrt{1 - 4y^2}}{y}$$

> **inverses[2];**

$$\frac{1}{2} \frac{1 - \sqrt{1 - 4y^2}}{y}$$

I want to make the inverses into operators. The command **unapply** converts an expression into an operator. The second argument to **unapply** is the variable that appears in the expression that you want to make into the parameter of the operator.

> **g1:= unapply(inverses[1], y);**

$$g1 := y \rightarrow \frac{1}{2} \frac{1 + \sqrt{1 - 4y^2}}{y}$$

> **g2:= unapply(inverses[2], y);**

$$g2 := y \rightarrow \frac{1}{2} \frac{1 - \sqrt{1 - 4y^2}}{y}$$

The two commands above are essentially the same, and I hate having to repeat the same command. The **map** command will take a command and apply to every member of a set or list.

> **g:= map(unapply, inverses, y);**

$$g := \left(y \rightarrow \frac{1}{2} \frac{1 + \sqrt{1 - 4y^2}}{y}, y \rightarrow \frac{1}{2} \frac{1 - \sqrt{1 - 4y^2}}{y} \right)$$

Separate the individual members:

> **g1:= g[1]; g2:= g[2];**

$$g1 := y \rightarrow \frac{1}{2} \frac{1 + \sqrt{1 - 4y^2}}{y}$$

$$g2 := y \rightarrow \frac{1}{2} \frac{1 - \sqrt{1 - 4y^2}}{y}$$

The separation can also be done with the single command

> **g1,g2:= g[];**

$$g1, g2 := y \rightarrow \frac{1}{2} \frac{1 + \sqrt{1 - 4y^2}}{y}, y \rightarrow \frac{1}{2} \frac{1 - \sqrt{1 - 4y^2}}{y}$$

Now we need to figure out which inverse goes with which domain. How could such a thing be determined? The range of an inverse is the domain the original function. The line $y = \frac{1}{4}$ crosses the graph of f twice, so we evaluate the inverses there.

> **g1(1/4); g2(1/4);**

$$2 + \frac{1}{2}\sqrt{3}\sqrt{4}$$

$$2 - \frac{1}{2}\sqrt{3}\sqrt{4}$$

Note that those commands can be combined as

```
> [g1, g2](1/4);
```

$$\left[2 + \frac{1}{2}\sqrt{3}\sqrt{4}, 2 - \frac{1}{2}\sqrt{3}\sqrt{4} \right]$$

It is not clear which intervals each of those is in. The command **evalf** converts numeric expressions into decimal form.

```
> evalf(%);
```

$$[3.732050808, 0.267949192]$$

Note that **evalf** automatically mapped over the list.

As I am writing this, I got $g_1(1/4)=3.73\dots$ and $g_2(1/4)=0.26\dots$. The definitions of g_1 and g_2 might be reversed for you. You cannot guarantee the order that solve will return things. The following command will ensure that we are talking about the same order.

Briefly, the command below says that if $g_1(1/4)$ is between a and b (-1 and 1), then g_1 and g_2 should be swapped. Note that lines beginning with **#** are comments rather than commands. The reason for the **eval**'s is that operator names are not automatically expanded to their definitions. Note that **fi** (which is **if** spelled backwards) denotes the end of the **if** command. You can also say **end if** or simply **end**, but I prefer the briefer notation.

```
> if is(a <= g1(1/4) and is(g1(1/4) <= b) then  
  # Swap g1 and g2  
  temp:= eval(g1); g1:= eval(g2); g2:= eval(temp)  
fi;
```

At this point, I am sure that g_2 corresponds to the f domain $[-1,1]$, and g_1 corresponds to the other domain.

```
> g2(x);
```

$$\frac{1}{2} \frac{1 - \sqrt{1 - 4x^2}}{x}$$

Note that g_2 is not defined at 0, yet 0 is in the corresponding f domain $[-1,1]$. That is because if $y=0$, the equation $\mathcal{Y} = f(x)$ is not a true quadratic equation (try it by hand and see for yourself), and hence the quadratic formula that the solve command used is not valid for that point. But the Inverse Function Theorem says that g_2 must be continuous. However, if we just *define* $g_2(0)$ to be 0, everything will be okay. This makes g_2 continuous, as you can see from the following limit:

```
> limit(g2(x), x= 0);
```

0

Note that in Maple's limit command the word "limit" is spelled out, and $x \rightarrow 0$ is indicated by the `x=0`.

There is no real need in this problem to tell Maple that $g2(0)=0$, but I might as well show you how it's done anyway. It's very simple:

```
> g2(0):= 0;
```

$$g2(0) := 0$$

Compute the tangent lines to f at selected points in each interval

I'll use the points -3/2, 1/2, 3/2.

```
> X:= [-3/2, 1/2, 3/2];
```

The tangent line at 1/2 can be expressed as an operator as follows:

```
> Tf:= D(f)(1/2)*(x-1/2)+f(1/2);
```

$$Tf := \frac{12}{25}x + \frac{4}{25}$$

We can generalize the procedure for finding tangent lines. Write an operator that takes a function and a specific point and returns the tangent line at that point.

```
> tangent_line:= (h,x0)-> D(h)(x0)*('x'-x0) + h(x0);
```

$$tangent_line := (h, x0) \rightarrow D(h)(x0) ('x' - x0) + h(x0)$$

Note that for an operator with two or more parameters, the parameters are put in parentheses before the arrow. The forward quotes on the `x` ensure that the above will work even if the variable `x` has already been assigned a value.

Test.

```
> Tf:= tangent_line(f,1/2);
```

$$Tf := \frac{12}{25}x + \frac{4}{25}$$

That's what we wanted.

Note that the expression $D(h)(x0)$ means to evaluate the function $D(h)$ at the point $x0$. That's not the same as $D(h(x0))$.

Now I want the tangent lines for all three values of $x0$. Remember **map** from above? Its first argument is an operator.

```
> Tf:= map(x0-> tangent_line(f,x0), X);
```

$$Tf := \left[-\frac{20}{169}x - \frac{108}{169}, \frac{12}{25}x + \frac{4}{25}, -\frac{20}{169}x + \frac{108}{169} \right]$$

Find the corresponding tangent lines to the inverses. Verify the derivatives predicted by the Inverse Function Theorem.

Now that I have given a general tangent_line procedure, this is easy. The points $X[1] = -3/2$ and $X[3] = 3/2$ are in the range of g_1 , and $X[2] = 1/2$ is in the range of g_2

> **Tg:= tangent_line(g1,f(X[1]));**

$$Tg := \left(-\frac{169}{72} - \frac{169}{1800} \sqrt{25} \sqrt{169} \right) \left(x + \frac{6}{13} \right) - \frac{13}{12} - \frac{1}{156} \sqrt{25} \sqrt{169}$$

That's not what I expected. I'll try to simplify it.

> **simplify(Tg);**

$$-\frac{169}{20}x - \frac{27}{5}$$

That's better. You can see that the slopes of the tangent lines are reciprocals.

The fact that I needed to simplify indicates that I should have simplified right in the definition of the procedure tangent_line. So, I will redefine tangent_line:

> **tangent_line:= (h,x0)-> simplify(D(h)(x0)*('x'-x0) + h(x0));**

$$\text{tangent_line} := (h, x0) \rightarrow \text{simplify}(D(h)(x0) ('x' - x0) + h(x0))$$

Better test the new procedure:

> **tangent_line(f,X[1]);**

$$-\frac{20}{169}x - \frac{108}{169}$$

> **Tg:= tangent_line(g1,f(X[1]));**

$$Tg := -\frac{169}{20}x - \frac{27}{5}$$

It works.

By the Inverse Function Theorem, we should have $g_1'(y) = 1/f'(g_1(y))$. That's clearly the case for our chosen point $y = f(-3/2)$, but I'll have Maple verify it for arbitrary points.

> **D(g1)(y);**

$$-\frac{1}{2} \frac{1 + \sqrt{1 - 4y^2}}{y^2} - \frac{2}{\sqrt{1 - 4y^2}}$$

> **1/D(f(g1)(y));**

$$\frac{1}{\frac{1}{\frac{1}{4} \frac{(1 + \sqrt{1 - 4y^2})^2}{y^2} + 1} - \frac{1}{2} \frac{(1 + \sqrt{1 - 4y^2})^2}{y^2 \left(\frac{1}{4} \frac{(1 + \sqrt{1 - 4y^2})^2}{y^2} + 1 \right)^2}}$$

Are they the same? It's hard to tell. One way to get Maple to verify that two things are equal is to subtract the two things, simplify the resulting expression, and see if you get zero.

> **simplify(% - %%);**

0

So they're equal. Note that %% refers to the second-most-recently-computed expression. There's also %%%, but you can't go back four or more levels.

Let's get all three corresponding inverse tangent lines.

> **Tg:= [tangent_line(g1, f(X[1])), tangent_line(g2, f(X[2])), tangent_line(g1, f(X[3]))];**

$$Tg := \left[-\frac{169}{20}x - \frac{27}{5}, \frac{25}{12}x - \frac{1}{3}, -\frac{169}{20}x + \frac{27}{5} \right]$$

Once again, I'd like to avoid needless repetition. The **zip** command is like **map**, but is used for two-argument operators.

> **Tg:= zip(tangent_line, [g1,g2,g1], map(f,X));**

$$Tg := \left[-\frac{169}{20}x - \frac{27}{5}, \frac{25}{12}x - \frac{1}{3}, -\frac{169}{20}x + \frac{27}{5} \right]$$

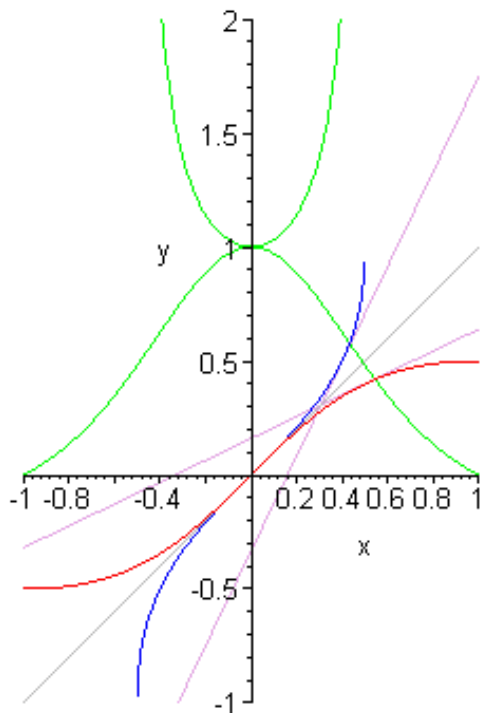
Plot it all

First I'll plot the parts corresponding to the central interval [-1,1]. Note that the order of colors in the colorlist corresponds to the order of the expressions in the expression list. In a list (things that are in square brackets), the order is significant. Any item in a list can be repeated any number of times by using the '\$'.

```

> colorlist:= [red, blue, grey, green$2, plum$2]:
> plot([f(x), g2(x), x, Df, D(g2)(x), Tf[2], Tg[2]]
,x= a..b, y= -1..2
,scaling=constrained
,color= colorlist
);

```



So the blue is f , red is the inverse, the lower green is f' , upper green is the derivative of the inverse, and the plums are the tangent lines. I want to save that plot in a variable so I can combine with other plots later. Note that I extended the y-range of the plot to 2 in order to include the derivative of the inverse.

```

> Plot1:= %:

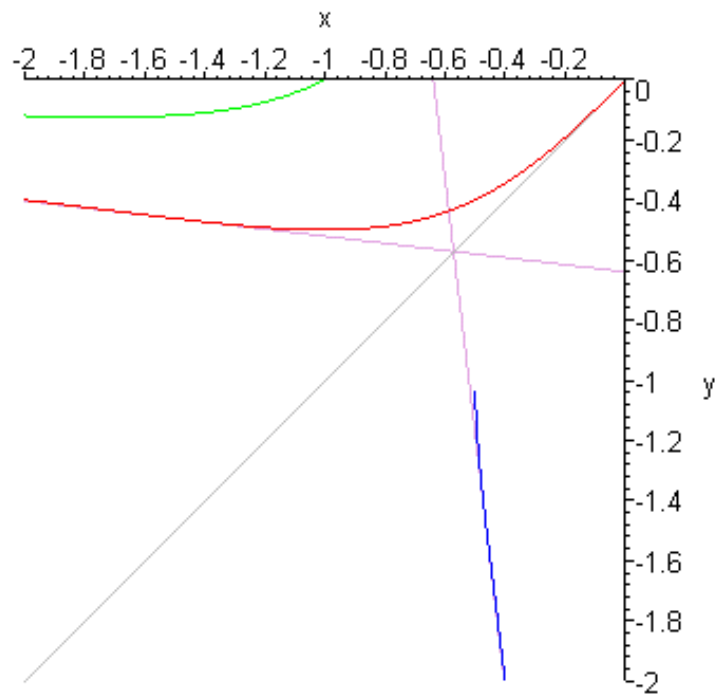
```

Now let's do the same for the left interval. I'll restrict it to $[-2,-1]$.

```

> plot([f(x), g1(x), x, Df, D(g1)(x), Tf[1], Tg[1]]
,x= -2..0, y= -2..0
,scaling=constrained
,color= colorlist);

```

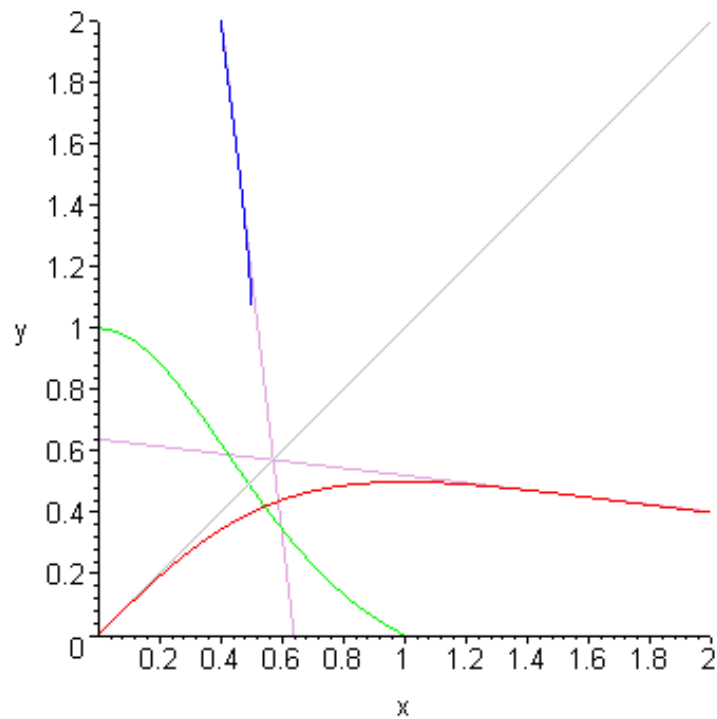


Note that I have to extend the plotting range in order to get the inverse in the picture.

> **Plot2:= %:**

Do the right interval.

> **plot([f(x), g1(x), x, Df, D(g1)(x), Tf[3], Tg[3]]
, x= 0..2, y= 0..2
, scaling= constrained
, color= colorlist);**



> **Plot3:= %:**

Now I want to put the three plots together. The command for that is **display**. Maple has many thousands of commands, and they are not all immediately available when Maple is started. They are grouped into sets called packages. The package **plots** contains **display**. The **with** command "loads" a package so that all of the package's commands are available.

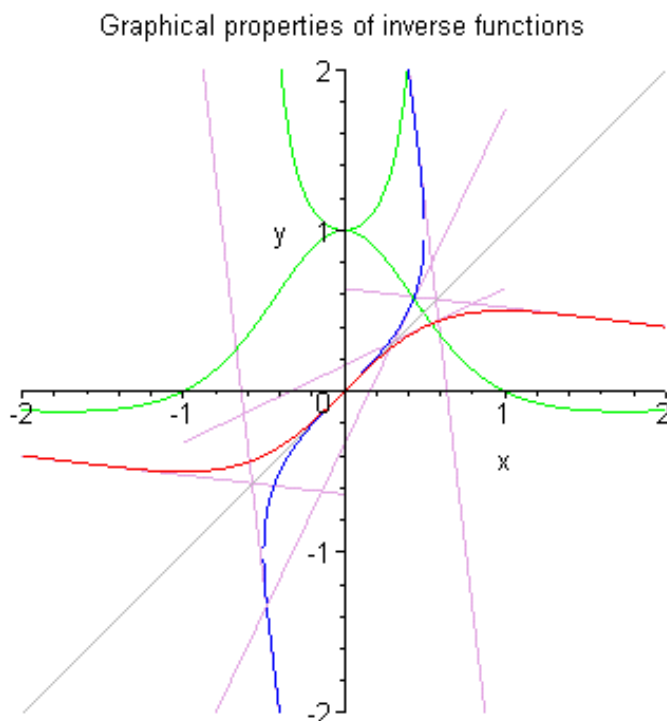
> **with(plots);**

Warning, the name `changecoords` has been redefined

[*animate, animate3d, animatecurve, arrow, changecoords, complexplot, complexplot3d, conformal, conformal3d, contourplot, contourplot3d, coordplot, coordplot3d, cylinderplot, densityplot, display, display3d, fieldplot, fieldplot3d, gradplot, gradplot3d, graphplot3d, implicitplot, implicitplot3d, inequal, interactive, listcontplot, listcontplot3d, listdensityplot, listplot, listplot3d, loglogplot, logplot, matrixplot, odeplot, pareto, plotcompare, pointplot, pointplot3d, polarplot, polygonplot, polygonplot3d, polyhedra_supported, polyhedraplot, replot, rootlocus, semilogplot, setoptions, setoptions3d, spacecurve, sparsematrixplot, sphereplot, surfdata, textplot, textplot3d, tubeplot*]

The above list shows the new commands that are now available. In the next command, `Plot||(1..3)` is a shorthand for Plot1, Plot2, Plot3.

> **display(Plot||(1..3), title= `Graphical properties of inverse functions`);**



The blue parts should be connected, but Maple has some difficulty plotting parts of a curve where there is a vertical tangent. I could correct this, but that is a little too complicated to go into right now. Note the reciprocal relationship of the two derivatives. The negative parts of the derivative of the inverse are so far into the negative range that it would distort the plot too much if I tried to include them.

Find a point where corresponding tangent lines intersect

> **ans:= solve(Tf[1]=Tg[1], x);**

$$ans := \frac{-4}{7}$$

We can get y-value of the point of intersection by evaluating either line at that x.

> **eval(Tf[1], x= ans);**

$$\frac{-4}{7}$$

So these tangent lines intersect on the line $y=x$. Intuition suggests that this will always be the case. Let's see if we can prove that with Maple.

Prove a simple, yet general, theorem with Maple

Theorem: Let f be a function which is injective on an interval and differentiable at a point x_0 in that interval. Let g be the inverse of f . Let L_1 be the tangent line to f at x_0 and let L_2 be the tangent line to g at $x = f(x_0)$. Then either L_1 and L_2 intersect on the line $y = x$, or $f'(x_0) = 1$, or $f'(x_0) = 0$.

Proof:

> **restart;**

Now all previous variable definitions are erased.

We take f as a completely arbitrary function and define g as its inverse:

> **g:= unapply(solve(y=f(x), x), y);**

$$g := y \rightarrow \text{RootOf}(f(_Z) - y)$$

RootOf is Maple's way of expressing the solution to an equation that it does not know how to solve. (**RootOf** can mean some other things also.) Of course, the equation $y = f(x)$ cannot be explicitly solved if you don't know what f is. However, even if Maple cannot not explicitly solve an equation, it can still know some things about the solution:

> **D(g);**

$$y \rightarrow \frac{1}{D(f)(\text{RootOf}(f(_Z) - y))}$$

That shows that Maple knows the Inverse Function Theorem.

> **tangent_line := (h,a)-> D(h)(a)*(x-a)+h(a):**

> **L[1] := tangent_line(f, x0);**

$$L_1 := D(f)(x_0) (x - x_0) + f(x_0)$$

> **L[2] := tangent_line(g, f(x0));**

$$L_2 := \frac{x - f(x_0)}{D(f)(\text{RootOf}(-f_Z) + f(x_0)))} + \text{RootOf}(-f_Z) + f(x_0)$$

If f is injective, the only root of the equation $f_Z - f(x_0) = 0$ is $Z = x_0$. (Recall that injective means that f can be cancelled from the equation $f(x) = f(y)$.) If $f'(x_0) = 0$, then of course the above tangent line is invalid.

> **RootOf(f_Z-f(x0)) := x0:**

Get the x-coordinate of the intersection of the tangents.

> **x_int := solve(L[1]=L[2], x);**

$$x_int := \frac{(-f(x_0) + D(f)(x_0) D(f)(\text{RootOf}(-f_Z) + f(x_0))) x_0 - f(x_0) D(f)(\text{RootOf}(-f_Z) + f(x_0))}{+ \text{RootOf}(-f_Z) + f(x_0) D(f)(\text{RootOf}(-f_Z) + f(x_0))) / (D(f)(x_0) D(f)(\text{RootOf}(-f_Z) + f(x_0))) - 1}$$

If $f'(x_0) = 1$, then the above is invalid. But the cases $f'(x_0) = 0$ and $f'(x_0) = 1$ are excluded in the conclusion of the theorem. Get the y-coordinate of the intersection.

> **y_int := subs(x= x_int, L[1]);**

$$y_int := D(f)(x_0) ((-f(x_0) + D(f)(x_0) D(f)(\text{RootOf}(-f_Z) + f(x_0))) x_0 - f(x_0) D(f)(\text{RootOf}(-f_Z) + f(x_0)) + \text{RootOf}(-f_Z) + f(x_0) D(f)(\text{RootOf}(-f_Z) + f(x_0))) / (D(f)(x_0) D(f)(\text{RootOf}(-f_Z) + f(x_0))) - 1) - x_0 + f(x_0)$$

> **simplify(x_int - y_int);**

$$\frac{(D(f)(x_0) D(f)(\text{RootOf}(-f_Z) + f(x_0))) x_0 - f(x_0) D(f)(\text{RootOf}(-f_Z) + f(x_0)) + \text{RootOf}(-f_Z) + f(x_0) D(f)(\text{RootOf}(-f_Z) + f(x_0)) + D(f)(x_0) f(x_0) - D(f)(x_0) \text{RootOf}(-f_Z) + f(x_0) D(f)(\text{RootOf}(-f_Z) + f(x_0)) - D(f)(x_0) x_0) / (D(f)(x_0) D(f)(\text{RootOf}(-f_Z) + f(x_0))) - 1)$$

Therefore, $x_int = y_int$; in other words, the point of intersection is on the line $y = x$.

Note again the technique of proving equality by simplifying the difference to 0.

Generalize inverse selection with RootOf

After we solved the equation $y = \frac{x}{x^2 + 1}$ for x , there were still some complicated steps involved to select the inverse. This

selection process can be simplified with **RootOf** at the expense of a more complicated representation of the inverse. Often, **solve** is only able to express the solutions in terms of **RootOf**.

> **f:= x-> x/(x^2+1);**

$$f := x \rightarrow \frac{x}{x^2 + 1}$$

We can specify the branch of the inverse with range $[-1,1]$ as follows:

> **g:= unapply(RootOf(y=f(x), x, -1..1), y);**

$$g := y \rightarrow \text{RootOf}(y_Z^2 + y - _Z, -1 .. 1)$$

Since Maple knows how to take derivatives of **RootOfs**, we can still get the tangent line.

> **tangent_line(g,f(1/2));**

$$\frac{\left(-\text{RootOf}(2_Z^2 + 2 - 5_Z, -1 .. 1)^2 - 1\right)\left(x - \frac{2}{5}\right)}{\frac{4}{5}\text{RootOf}(2_Z^2 + 2 - 5_Z, -1 .. 1) - 1} + \text{RootOf}(2_Z^2 + 2 - 5_Z, -1 .. 1)$$

But the roots are not computed until you use **allvalues**.

> **allvalues(%);**

$$\frac{25}{12}x - \frac{1}{3}$$

evalf also works on **RootOf**:

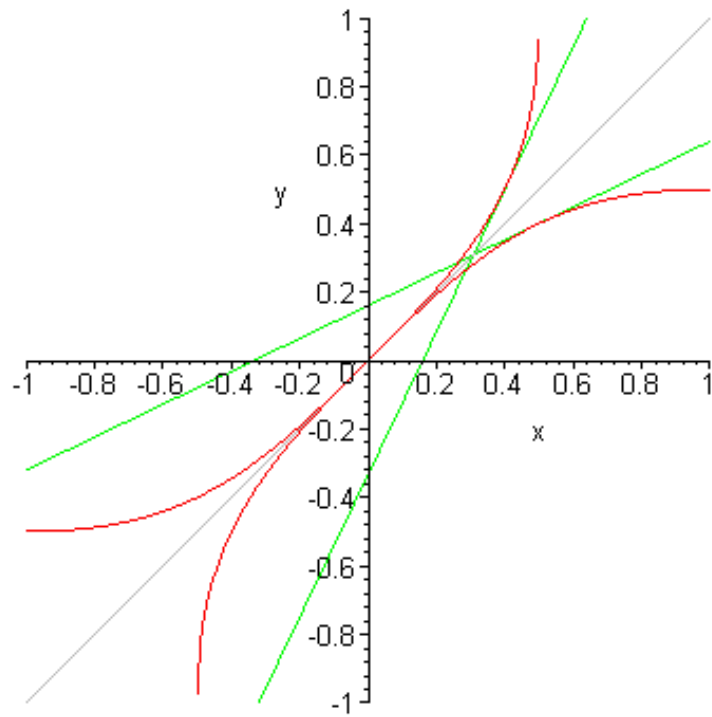
> **g(1/2);**

$$\text{RootOf}(_Z^2 + 1 - 2_Z, -1 .. 1)$$

> **evalf(%);**

$$1.000000000$$

```
> plot([f(x),g(x),x,tangent_line(f,1/2),allvalues(tangent_line(g,f(1/2)))]  
      ,x= -1..1, y= -1..1  
      ,scaling= constrained  
      ,color= [red$2, grey, green$2]  
      );
```



>

>