Re: how to solve linear curve fitting by casting it as a minimization problem

---

Source: http://sci.tech−archive.net/Archive/sci.math.num−analysis/2008−08/msg00219.html

---

- *From*: "eyh5@xxxxxxxxxxxxxxx" <eyh5@xxxxxxxxxxxxxxx>
- *Date*: Sun, 24 Aug 2008 10:13:34 −0700 (PDT)

On Aug 24, 7:13 am, none <n...@xxxxxxxx> wrote:

> On Sat, 23 Aug 2008 23:35:06 −0700, e...@xxxxxxxxxxxxxxx wrote:
>
>> Also to add: for simplicity, let's say that GA>GB>GC>GD.
>>
>> On Aug 24, 2:31 am, "e...@xxxxxxxxxxxxxxx" <e...@xxxxxxxxxxxxxxx>
>> wrote:
>>
>>> Hello,
>>>
>>> I'm wondering if anybody can help me with this problem.
>>> Let's first
>>> limit ourselves to only the first quadrant of a Cartesian
>>> coordinate
>>> system. I have a point G that's located on the x−axis, and it's
>>> position (x_G, 0) is known (x_G>0). Now, I have four points
>>> A, B, C,
>>> and D in this quadrant. I know these points are on a straight
>>> line
>>> (call it AD). This line has some arbitrary slope. I also know
>>> the four
>>> distances GA, GB, GC, and GD. Moreover, these four
>>> distances are
>>> periodically sampled, meaning that AB=BC=CD. However, I
>>> do not know
>>> the coordinates of these four points in the Cartesian system
>>> (denoted
>>> as (x_A, y_A), (x_B, y_B), (x_C, y_C), (x_D, y_D),
>>> respectively).

> > I want to find the slope of the line AD in the Cartesian
> > system.
> > The way I'm thinking about doing is to cast this as a
> > minimization
> > problem to find the points' coordinates. That way, the
> > traejctory
> > slope is simply = (y_A − y_B) / (x_A − x_B).
> >
> > However, I'm not sure how to formulate the minimization
> > (what
> > should be the objective function, and what should be the
> > constraint
> > function). Can anyone help me with this? Thanks in advance.

> Is this a homeowrk problem? To me it seems that if you know
>
> (x0, y0)
> slope m in y = m x + c −> c can be calculated.
> Simpler would be to work in a transformed coordinate so that your
> working space is centred on (0, 0) −> c = 0
>
> and spacing Delta−l = constant & known −> spacing Delta x = constant &
> unknown
>
> then all that remains is simple trigonometry.
>
> sin theta = Delta x/ Delta l would be useful.
>
> You might want to remember that the slope, m, is given by Delta y/Delta x,
> and that tan theta = Delta y/Delta x.

Thanks for the reply. No, this is not a homework problem. It's a part
of a bigger problem that I'm trying to solve in my research work.

I was able to do it in a transformed Cartesian system (with GA as the
x−axis, and a y−axis perpendicular to it; this gets rid of the
intercept, c, as you suggested), and trigonometry was indeed used to
transform (or translate) it into the slope in the original Cartesian
system. However, the solution is not accepted and my adviser is
pushing to solve it by linear curve fitting. And I cannot think of
another approach except doing it by means of some minimization
problem, which is why I posted this message in the first place.

Also, by (x0, y0), I take it you mean the coordinates of point A?
Unfortunately, that knowledge is assumed unknown to me. I don't know
the coordinates of any of the points A, B, C, and D. I agree that
having the knowledge of A's coordinates would help a lot. But the lack

of it is what makes this problem seem so difficult.

I'm looking at the problem again and am now wondering: if all the information I know is the four distance measurements that I can measure (periodically) from a single source G on the x−axis of a global Cartesian system, does that seem to suggest that it could have an infinite number of solutions for the slope? Because we can basically rotate all four measurements simultaneously around G, and that induces infinites solutions. Would you agree?

I appreciate your further comments.
.