Geodesic mid-point for arc-length

This code uses the results of `geodesic-lsq` and `metric` to show that the 2nd and 3rd order estimates for L^2_{PQ} can be recovered using a mid-point estimate. For the 3rd order estimate we have

$$
\begin{aligned}
g_{ab}(x) &= g_{ab} - \frac{1}{3}x^c x^d R_{acbd} - \frac{1}{6}x^c x^d x^e \nabla_c R_{adbe} + \mathcal{O}\left(\epsilon^4\right) \\
L^2_{PQ} &= g_{ab} Dx^a Dx^b - \frac{1}{3}x^a x^b Dx^c Dx^d R_{acbd} - \frac{1}{12}x^a x^b Dx^c Dx^d Dx^e \nabla_c R_{adbe} - \frac{1}{6}x^a x^b x^c Dx^d Dx^e \nabla_a R_{bdce} + \mathcal{O}\left(\epsilon^4\right)
\end{aligned}
$$

The code below verifies that

$$L^2_{PQ} = g_{ab}(\bar{x}) Dx^a Dx^b + \mathcal{O}\left(\epsilon^4\right)$$

where $\bar{x}$ is the *coordinate* midpoint of the geodesic

$$\bar{x}^a = \frac{1}{2}\left(x^a_P + x^a_Q\right)$$

This result holds true only for the 2nd and 3rd order estimates. Note that the *coordinate* midpoint is not the *geometric* midpoint of the geodesic.

It might be interesting to see if the higher order estimates could be recovered by sampling the metric at points other than the mid point.

```
{a,b,c,d,e,f,g,h,i,j,k,l,m,n,o,p,q,r,s,t,u,v,w#}::Indices(position=independent).

\nabla{#}::Derivative.

g_{a b}::Metric.
g^{a b}::InverseMetric.

R_{a b c d}::RiemannTensor.

import cdblib

gab = cdblib.get('g_ab','metric.json')

lsq2 = cdblib.get('lsq2','geodesic-lsq.json')
lsq3 = cdblib.get('lsq3','geodesic-lsq.json')
lsq4 = cdblib.get('lsq4','geodesic-lsq.json')
lsq5 = cdblib.get('lsq5','geodesic-lsq.json')

substitute (gab,$x^{a}->(p^{a}+q^{a})/2$)    # evaluate rnc gab at mid-point
distribute (gab)

defgab := g_{a b} -> @(gab).

mid := g_{a b} (q^{a}-p^{a}) (q^{b}-p^{b}).

substitute     (mid,defgab)
distribute     (mid)
sort_product   (mid)
rename_dummies (mid)
canonicalise   (mid)

tst2 := @(lsq2) - @(mid).                         # cdb (tst2.201,tst2)
tst3 := @(lsq3) - @(mid).                         # cdb (tst3.201,tst3)
tst4 := @(lsq4) - @(mid).                         # cdb (tst4.201,tst4)
tst5 := @(lsq5) - @(mid).                         # cdb (tst5.201,tst5)

substitute     (tst2,$Dx^{a} -> q^{a}-p^{a}$)
substitute     (tst2,$x^{a} -> p^{a}$)
```

```
distribute      (tst2)
sort_product    (tst2)
rename_dummies  (tst2)
canonicalise    (tst2)                              # cdb (tst2.202,tst2)

substitute      (tst3,$Dx^{a} -> q^{a}-p^{a}$)
substitute      (tst3,$x^{a} -> p^{a}$)
distribute      (tst3)
sort_product    (tst3)
rename_dummies  (tst3)
canonicalise    (tst3)                              # cdb (tst3.202,tst3)

substitute      (tst4,$Dx^{a} -> q^{a}-p^{a}$)
substitute      (tst4,$x^{a} -> p^{a}$)
distribute      (tst4)
sort_product    (tst4)
rename_dummies  (tst4)
canonicalise    (tst4)                              # cdb (tst4.202,tst4)

substitute      (tst5,$Dx^{a} -> q^{a}-p^{a}$)
substitute      (tst5,$x^{a} -> p^{a}$)
distribute      (tst5)
sort_product    (tst5)
rename_dummies  (tst5)
canonicalise    (tst5)                              # cdb (tst5.202,tst5)
```

Reformatting

```
def truncateR (obj,n):

# I would like to assign different weights to \nabla_{a}, \nabla_{a b}, \nabla_{a b c} etc. but no matter
# what I do it appears that Cadabra assigns the same weight to all of these regardless of the number of subscripts.
# It seems that the weight is assigned to the symbol \nabla alone. So I'm forced to use the following substitution trick.

    Q_{a b c d}::Weight(label=numR,value=2).
    Q_{a b c d e}::Weight(label=numR,value=3).
    Q_{a b c d e f}::Weight(label=numR,value=4).
    Q_{a b c d e f g}::Weight(label=numR,value=5).

    tmp := @(obj).

    substitute (tmp, $\nabla_{e f g}{R_{a b c d}} -> Q_{a b c d e f g}$)
    substitute (tmp, $\nabla_{e f}{R_{a b c d}} -> Q_{a b c d e f}$)
    substitute (tmp, $\nabla_{e}{R_{a b c d}} -> Q_{a b c d e}$)
    substitute (tmp, $R_{a b c d} -> Q_{a b c d}$)

    ans = Ex(0)

    for i in range (0,n+1):
       foo := @(tmp).
       bah = Ex("numR = " + str(i))
       keep_weight (foo, bah)
       ans = ans + foo

    substitute (ans, $Q_{a b c d e f g} -> \nabla_{e f g}{R_{a b c d}}$)
    substitute (ans, $Q_{a b c d e f} -> \nabla_{e f}{R_{a b c d}}$)
    substitute (ans, $Q_{a b c d e} -> \nabla_{e}{R_{a b c d}}$)
    substitute (ans, $Q_{a b c d} -> R_{a b c d}$)

    return ans

tst2 = truncateR (tst2,2)  # cdb (tst2.301,tst2)
tst3 = truncateR (tst3,3)  # cdb (tst3.301,tst3)
tst4 = truncateR (tst4,4)  # cdb (tst4.301,tst4)
```

```
tst5 = truncateR (tst5,5)  # cdb (tst5.301,tst5)
```

Errors is mid-point estimates for L^2_{PQ}

$$\left(L^2_{PQ} - g_{ab}(\bar{x})Dx^aDx^b\right)_2 = 0$$

$$\left(L^2_{PQ} - g_{ab}(\bar{x})Dx^aDx^b\right)_3 = 0$$

$$\begin{aligned}\left(L^2_{PQ} - g_{ab}(\bar{x})Dx^aDx^b\right)_4 = &-\frac{1}{30}R_{abcd}R_{efgh}g^{ae}p^cp^gq^bq^dq^fq^h + \frac{1}{15}R_{abcd}R_{efgh}g^{ae}p^bp^cp^gq^dq^fq^h - \frac{1}{30}R_{abcd}R_{efgh}g^{ae}p^bp^cp^fp^gq^dq^h\\
&+\frac{1}{240}\nabla_{ab}R_{cdef}p^bp^cp^eq^aq^dq^f - \frac{1}{240}\nabla_{ab}R_{cdef}p^ap^bp^cp^eq^dq^f + \frac{1}{240}\nabla_{ab}R_{cdef}p^ap^cp^eq^bq^dq^f - \frac{1}{240}\nabla_{ab}R_{cdef}p^cp^eq^aq^bq^dq^f\end{aligned}$$

$$\begin{aligned}\left(L^2_{PQ} - g_{ab}(\bar{x})Dx^aDx^b\right)_5 = &-\frac{1}{30}R_{abcd}R_{efgh}g^{ae}p^cp^gq^bq^dq^fq^h + \frac{1}{15}R_{abcd}R_{efgh}g^{ae}p^bp^cp^gq^dq^fq^h - \frac{1}{30}R_{abcd}R_{efgh}g^{ae}p^bp^cp^fp^gq^dq^h\\
&+\frac{1}{240}\nabla_{ab}R_{cdef}p^bp^cp^eq^aq^dq^f - \frac{1}{240}\nabla_{ab}R_{cdef}p^ap^bp^cp^eq^dq^f + \frac{1}{240}\nabla_{ab}R_{cdef}p^ap^cp^eq^bq^dq^f - \frac{1}{240}\nabla_{ab}R_{cdef}p^cp^eq^aq^bq^dq^f\\
&+\frac{1}{135}R_{abcd}\nabla_eR_{fghi}g^{af}p^cp^gp^hq^bq^dq^eq^i + \frac{7}{270}R_{abcd}\nabla_eR_{fghi}g^{af}p^cp^ep^gp^hq^bq^dq^i - \frac{1}{90}R_{abcd}\nabla_eR_{fghi}g^{af}p^bp^cp^gp^hq^dq^eq^i\\
&-\frac{1}{45}R_{abcd}\nabla_eR_{fghi}g^{af}p^bp^cp^ep^gp^hq^dq^i - \frac{1}{90}R_{abcd}\nabla_eR_{fghi}g^{af}p^cp^ep^hq^bq^dq^gq^i + \frac{1}{135}R_{abcd}\nabla_eR_{fghi}g^{af}p^bp^cp^ep^hq^dq^gq^i\\
&-\frac{1}{108}R_{abcd}\nabla_eR_{fghi}g^{ae}p^cp^fp^hq^bq^dq^gq^i + \frac{1}{108}R_{abcd}\nabla_eR_{fghi}g^{ae}p^bp^cp^fp^hq^dq^gq^i - \frac{1}{45}R_{abcd}\nabla_eR_{fghi}g^{af}p^cp^hq^bq^dq^eq^gq^i\\
&+\frac{7}{270}R_{abcd}\nabla_eR_{fghi}g^{af}p^bp^cp^hq^dq^eq^gq^i + \frac{1}{2160}\nabla_{abc}R_{defg}p^bp^cp^dp^fq^aq^eq^g - \frac{1}{720}\nabla_{abc}R_{defg}p^ap^bp^cp^dp^fq^eq^g\\
&+\frac{1}{2160}\nabla_{abc}R_{defg}p^ap^cp^dp^fq^bq^eq^g + \frac{1}{2160}\nabla_{abc}R_{defg}p^ap^bp^dp^fq^cq^eq^g + \frac{1}{2160}\nabla_{abc}R_{defg}p^cp^dp^fq^aq^bq^eq^g\\
&+\frac{1}{2160}\nabla_{abc}R_{defg}p^bp^dp^fq^aq^cq^eq^g + \frac{1}{2160}\nabla_{abc}R_{defg}p^ap^dp^fq^bq^cq^eq^g - \frac{1}{720}\nabla_{abc}R_{defg}p^dp^fq^aq^bq^cq^eq^g\end{aligned}$$