Exercise 18

Please attempt all of the following problems before the due date. Your grade on this assignment will be calculated from the best three answers.

Problem 18.1

The text claims that, in the relation

\[ vf = v \cdot df \]

the right side of the equation is locally linear in the form \( df \) but the left side is not locally linear in the function \( f \). On the left side, \( v \) obeys Leibniz’s rule. On the right side, it does not. Multiply the function \( f \) by another function \( g \) and show that these properties are consistent. What is it that obeys Leibniz’s rule on the right side of the relation?

Answer 18.1

\[ v(fg) = gvf + fvg \]
\[ v(fg) = v \cdot d(fg) \]
\[ vf = v \cdot df \]
\[ vg = v \cdot dg \]

so Leibniz’s rule for \( v(fg) \) can also be written as

\[ v \cdot d(fg) = g(v \cdot df) + fv \cdot dg \]

Since everything on the right is locally linear, the functions \( g \) and \( f \) can be moved around a bit so this relation becomes

\[ v \cdot d(fg) = v \cdot gdf + v \cdot fdg \]

or, since \( v \) is arbitrary,

\[ d(fg) = gdf + fdg \]

and it is the \( d \) operator that obeys Leibniz’s rule on the right side of the relation.
Problem 18.2

Show that the direct construction of the covariant derivative of a tensor by subtracting counter terms for each argument as in the expression,

\[(D_v T)(P, a_1(P), a_2(P), \ldots, a_q(P)) = D_v (T(P, a_1(P), a_2(P), \ldots, a_q(P))) - T(P, D_v a_1(P), a_2(P), \ldots, a_q(P)) - T(P, a_1(P), D_v a_2(P), \ldots, a_q(P)) \ldots - T(P, a_1(P), a_2(P), \ldots, D_v a_q(P))\]

does, in fact, produce an expression that is locally linear in each of its tensor arguments.

(Hint: Use the local linearity of \(T\) together with Leibniz’s product rule for \(L_v\) acting on the product of two functions and for \(D_v\) acting on the product of a function and a tensor.)

Answer 18.2

It helps to suppress all the dependence on the evaluation point \(P\) so that we just have

\[(D_v T)(a_1, a_2, \ldots, a_q) = D_v (T(a_1, a_2, \ldots, a_q)) - T(D_v a_1, a_2, \ldots, a_q)
- T(a_1, D_v a_2, a_3, a_4) \ldots - T(a_1, a_2, \ldots, D_v a_q)\]

and the local linearity that we need to check is

\[(D_{f_v} T)(a_1, a_2, \ldots, a_q) = f(D_v T)(a_1, a_2, \ldots, a_q)\]

and

\[(D_v T)(fa_1, a_2, \ldots, a_q) = (D_v T)(a_1, fa_2, \ldots, a_q) = \ldots = (D_v T)(a_1, a_2, \ldots, fa_q) = f(D_v T)(a_1, a_2, \ldots, a_q)\]

Check the first relation, using the fact that \(D_v\) is defined so that \(D_{f_v} = fD_v\). To confirm that this definition is consistent with that requirement, calculate

\[(D_{f_v} T)(a_1, a_2, \ldots, a_q) = D_{f_v} (T(a_1, a_2, \ldots, a_q)) - T(D_{f_v} a_1, a_2, \ldots, a_q)
- T(a_1, D_{f_v} a_2, \ldots, a_q) \ldots - T(a_1, a_2, \ldots, D_{f_v} a_q)\]

Use \(D_{f_v} = fD_v\) for the action of \(D\) on the function \(T(a_1, a_2, \ldots, a_q)\) and on each of the tensors \(a_n\),

\[(D_{f_v} T)(a_1, a_2, \ldots, a_q) = fD_v (T(a_1, a_2, \ldots, a_q)) - T(fD_v a_1, a_2, \ldots, a_q)
- T(a_1, fD_v a_2, \ldots, a_q) \ldots - T(a_1, a_2, \ldots, fD_v a_q)\]

Now use the local linearity of the tensor \(T\) to obtain

\[(D_{f_v} T)(a_1, a_2, \ldots, a_q) = fD_v (T(a_1, a_2, \ldots, a_q)) - fT(D_v a_1, a_2, \ldots, a_q)
- DT(a_1, D_v a_2, \ldots, a_q) \ldots - DT(a_1, a_2, \ldots, D_v a_q)\]

\[= (fD_v T)(a_1, a_2, \ldots, a_q)\]

Now check the second series of relations. The first one is

\[(D_v T)(fa_1, a_2, \ldots, a_q)\]
\[ D_v (T (fa_1, a_2, \ldots, a_q)) - T (D_v (fa_1), a_2, \ldots, a_q) \]
\[ - T (fa_1, D_v a_2, \ldots, a_q) \ldots - T (fa_1, a_2, \ldots, D_v a_q) \]

Use the local linearity of \( T \) on three of these terms and Leibniz’s rule on \( D_v (fa_1) \).

\[(D_v T) (fa_1, a_2, \ldots, a_q) \]
\[= D_v fT (a_1, a_2, \ldots, a_q) - T ((D_v f) a_1 + fD_v a_1, a_2, \ldots, a_q) \]
\[ - fT (a_1, D_v a_2, \ldots, a_q) \ldots - fT (a_1, a_2, \ldots, D_v a_q) \]

Use the local linearity of \( T \) again and Leibniz’s rule on \( D_v (fT (a_1, a_2, \ldots, a_q)) \).

\[(D_v f) T (a_1, a_2, \ldots, a_q) + fD_v T (a_1, a_2, \ldots, a_q) - (D_v f) T (a_1, a_2, \ldots, a_q) \]
\[ - fT (D_v a_1, a_2, \ldots, a_q) \]
\[ - fT (a_1, D_v a_2, \ldots, a_q) \ldots - fT (a_1, a_2, \ldots, D_v a_q) \]

Notice that the terms with \((D_v f)\) cancel.

\[(D_v T) (fa_1, a_2, \ldots, a_q) \]
\[= fD_v T (a_1, a_2, \ldots, a_q) \]
\[ - fT (D_v a_1, a_2, \ldots, a_q) \]
\[ - fT (a_1, D_v a_2, \ldots, a_q) \ldots - fT (a_1, a_2, \ldots, D_v a_q) \]
\[= f (D_v T) (a_1, a_2, \ldots, a_q) \]

The same sequence of operations works on each of the arguments, so there is local linearity in all of them.
Problem 18.3

Show that the direct construction procedure described above will always give the same result as assuming that tensor products and dot products obey Leibniz’ rule.

(Hint: Show that it works for vector and form fields and then use induction.)

Answer 18.3

First use the direct construction procedure on a one-form \( \alpha \)

\[ (D_v \alpha) (u) = D_v (\alpha (u)) - \alpha (D_v u) \]

and notice that this can be re-arranged as

\[ D_v (\alpha (u)) = (D_v \alpha) (u) + \alpha (D_v u) \]

which can be written as Leibniz’s rule for the dot product:

\[ D_v (\alpha \cdot u) = D_v \alpha \cdot u + \alpha \cdot D_v u \]

Thus, for forms and vectors, the two requirements are the same.

Next, consider the tensor product of tensors \( S \) and \( T \). The tensor \( S \) assigns the number \( S(a_1, a_2, ..., a_q) \) to the forms and vectors \( a_1, a_2, ..., a_q \) while the tensor \( T \) assigns the number \( T(b_1, b_2, ..., b_r) \) to the forms and vectors \( b_1, b_2, ..., b_r \).

The tensor product \( S \otimes T \) assigns the number

\[ S \otimes T(a_1, a_2, ..., a_q, b_1, b_2, ..., b_r) = S(a_1, a_2, ..., a_q) T(b_1, b_2, ..., b_r) \]

to the forms and vectors \( a_1, a_2, ..., a_q, b_1, b_2, ..., b_r \).

The direct construction procedure for finding \( D_v (S \otimes T) \) is as follows:

\[ D_v(S \otimes T)(a_1, a_2, ..., a_q, b_1, b_2, ..., b_r) \]
\[ = D_v(S \otimes T)(a_1, a_2, ..., a_q) T(b_1, b_2, ..., b_r) \]
\[ - S \otimes T(D_v a_1, a_2, ..., a_q, b_1, b_2, ..., b_r) \]
\[ - S \otimes T(D_v a_2, a_3, ..., a_q, b_1, b_2, ..., b_r) - ... \]
\[ - S \otimes T(D_v a_q, a_1, a_2, ..., a_{q-1}, b_1, b_2, ..., b_r) \]
\[ - S \otimes T(a_1, a_2, ..., a_q, D_v b_1, b_2, ..., b_r) \]
\[ - S \otimes T(a_1, a_2, ..., a_q, b_1, D_v b_2, ..., b_r) - ... \]
\[ - S \otimes T(a_1, a_2, ..., a_q, b_1, b_2, ..., D_v b_r) \]

Use the definition of the tensor product to rewrite each term.

\[ D_v(S \otimes T)(a_1, a_2, ..., a_q, b_1, b_2, ..., b_r) \]
\[ = D_v(S(a_1, a_2, ..., a_q) T(b_1, b_2, ..., b_r)) \]
\[ - S(D_v a_1, a_2, ..., a_q) T(b_1, b_2, ..., b_r) \]
\[ - S(a_1, D_v a_2, ..., a_q) T(b_1, b_2, ..., b_r) - ... \]
\[ - S(a_1, a_2, ..., D_v a_q) T(b_1, b_2, ..., b_r) \]
\[ - S(a_1, a_2, ..., a_q) T(D_v b_1, b_2, ..., b_r) \]
\[ - S(a_1, a_2, ..., a_q) T(b_1, D_v b_2, ..., b_r) - ... \]
\[ - S(a_1, a_2, ..., a_q) T(b_1, b_2, ..., D_v b_r) \]

The first term is just the derivative of a product of two functions, so use Leibniz’s rule on it and then group together terms with common factors.

\[ D_v(S \otimes T)(a_1, a_2, ..., a_q, b_1, b_2, ..., b_r) \]
\[ = D_v S(a_1, a_2, ..., a_q) T(b_1, b_2, ..., b_r) + (S(a_1, a_2, ..., a_q) D_v T(b_1, b_2, ..., b_r)) \]
\[ - S(D_v a_1, a_2, ..., a_q) T(b_1, b_2, ..., b_r) \]
\[ - S(a_1, D_v a_2, ..., a_q) T(b_1, b_2, ..., b_r) - ... \]
\[ - S(a_1, a_2, ..., D_v a_q) T(b_1, b_2, ..., b_r) \]
\[-S(a_1, a_2, \ldots, a_q) T(D_v b_1, b_2, \ldots, b_r)\]
\[-S(a_1, a_2, \ldots, a_q) T(b_1, D_v b_2, \ldots, b_r) - \ldots\]
\[-S(a_1, a_2, \ldots, a_q) T(b_1, b_2, \ldots, D_v b_r)\]
\[= [D_v S(a_1, a_2, \ldots, a_q) - S(D_v a_1, a_2, \ldots, a_q) - S(a_1, D_v a_2, \ldots, a_q)\]
\[\ldots - S(a_1, a_2, \ldots, D_v a_q) T(b_1, b_2, \ldots, b_r)\]
\[+ S(a_1, a_2, \ldots, a_q) [D_v T(b_1, b_2, \ldots, b_r) - T(D_v b_1, b_2, \ldots, b_r) - T(b_1, D_v b_2, \ldots, b_r)\]
\[\ldots - T(b_1, b_2, \ldots, D_v b_r)]\]

The terms in square brackets are just the constructive definitions of the derivatives $D_v T$ and $D_v S$ so we get

$D_v (S \otimes T) (a_1, a_2, \ldots, a_q, b_1, b_2, \ldots, b_r) = (D_v S) (a_1, a_2, \ldots, a_q) T(b_1, b_2, \ldots, b_r) + S(a_1, a_2, \ldots, a_q) (D_v T)(b_1, b_2, \ldots, b_r)$

which can be written as

$D_v (S \otimes T) (a_1, a_2, \ldots, a_q, b_1, b_2, \ldots, b_r) = (D_v S) \otimes T (a_1, a_2, \ldots, a_q b_1, b_2, \ldots, b_r) + S \otimes D_v T (a_1, a_2, \ldots, a_q b_1, b_2, \ldots, b_r)$

or

$D_v (S \otimes T) = (D_v S) \otimes T + S \otimes D_v T$.

Thus, the constructive definition implies that all tensor products obey Leibniz’s rule.
Problem 18.4

Use the direct construction of the covariant derivative of a one-form field $\alpha$

$$(D_v \alpha) (u) = \nabla_v (\alpha (u)) - \alpha (\nabla_v u)$$

to obtain the result

$$D_v \alpha = v^d (e_d (\alpha_A) - \alpha_K \Gamma^{K}_{Ad}) W^A$$

obtained by applying Leibniz’s rule to the dot product.

Answer 18.4

Expand

$v = v^d e_d$

and

$\alpha = \alpha_A W^A$

and

$u = u^A E_A$

in the direct construction:

$$(D_v \alpha) (u) = \nabla_v (\alpha (u)) - \alpha (\nabla_v u)$$

$$= \nabla_v e_d (\alpha (u)) - \alpha (\nabla_v e_d u)$$

$$= v^d [\nabla_{e_d} (\alpha (u)) - \alpha (\nabla_{e_d} u)]$$

$$= v^d [\nabla_{e_d} (\alpha_A u^A) - \alpha (\nabla_{e_d} (u^A E_A))$$

$$= v^d [e_d (\alpha_A) u^A + \alpha A e_d (u^A) - \alpha (e_d (u^A) E_A) - \alpha (u^A \Gamma^{K}_{Ad} E_K)]$$

$$= v^d [e_d (\alpha_A) u^A + \alpha A e_d (u^A) - \alpha_K u^A \Gamma^{K}_{Ad}]$$

$$= v^d [e_d (\alpha_A) - \alpha_K \Gamma^{K}_{Ad}] u^A$$

$$= v^d [e_d (\alpha_A) - \alpha_K \Gamma^{K}_{Ad}] W^A (u)$$

Since $u$ is arbitrary, we get

$$D_v \alpha = v^d [e_d (\alpha_A) - \alpha_K \Gamma^{K}_{Ad}] W^A$$