Computational Geometry Workshop

Wichita State University, Dept. of Mathematics

Differential Forms

Last Changed: 4 Sep 2017

Authors: Justin M. Ryan,

This notebook contains sample code related to the lecture on Differential Forms. Terse lecture notes may be found at

http://geometerjustin.com/teaching/cgw/notes/notes_02.html (http://geometerjustin.com/teaching/cgw/notes/notes_02.html)

So future readers know what version we're using:

In [1]:
```
version()
```

Out[1]: 'SageMath version 7.5.1, Release Date: 2017-01-15'

Force $\LaTeX$ display:

In [2]:
```
%display latex
```

Define the chart we're working in/on:

In [3]:
```
U = Manifold(3,'U',latex_name=r'\mathbb{U}',start_index=1);
coord.<x,y,z>=U.chart();
e = coord.frame();
de = coord.coframe();
```

We want to work with the differential k-forms on $\mathbb{U}$. We begin by defining two 1-forms on $\mathbb{U}$.

In [4]:
```
t1 = U.diff_form(1,latex_name=r'\theta_1',);
t1[:] =[x, 2*z-x,sin(y)];
t2 = U.diff_form(1,latex_name=r'\theta_2');
t2[:] =[0,-z,y];
```

We can now compute the exterior product of θ_1 and θ_2. The command for this is just ".wedge()". The exterior product $\theta_1 \wedge \theta_2$ is a 2-form.

In [5]:
```
t1w2 = t1.wedge(t2);
print(t1w2)
t1w2.display()
```

```
2-form on the 3-dimensional differentiable manifold U
```

Out[5]: $$\theta_1 \wedge \theta_2 = -xz\,\mathrm{d}x \wedge \mathrm{d}y + xy\,\mathrm{d}x \wedge \mathrm{d}z + (-xy + (2\,y + \sin(y))z)\,\mathrm{d}y \wedge \mathrm{d}z$$

Exterior products are alternating.

In [6]:
```
t2w1 = t2.wedge(t1);
t2w1 == -t1w2
```

Out[6]: True

The exterior derivative of a form increases the degree by 1.

In [7]:
```
dt1 = t1.exterior_derivative();
dt2 = t2.exterior_derivative();
print(dt1)
dt1.display()
```

```
2-form on the 3-dimensional differentiable manifold U
```

Out[7]: $$\mathrm{d}\theta_1 = -\mathrm{d}x \wedge \mathrm{d}y + (\cos(y) - 2)\,\mathrm{d}y \wedge \mathrm{d}z$$

In [8]:
```
print(dt2)
dt2.display()
```

```
2-form on the 3-dimensional differentiable manifold U
```

Out[8]: $$\mathrm{d}\theta_2 = 2\mathrm{d}y \wedge \mathrm{d}z$$

The exterior derivative of a smooth function $f \in \mathfrak{F} = \Omega^0$ is the usual differential of f from Calc III. Recall that we need to define smooth functions on $\mathbb{U}$ as scalar fields.

In [9]:
```
f = U.scalar_field({coord: x^2 - 2*z^(sin(y))});
df = f.exterior_derivative();
print(df)
df.display()
```

```
1-form on the 3-dimensional differentiable manifold U
```

Out[9]: $2\,x\mathrm{d}x - 2\,z^{\sin(y)}\cos(y)\log(z)\mathrm{d}y + \left(-2\,z^{\sin(y)-1}\sin(y)\right)\mathrm{d}z$

In [10]:
```
df == f.differential()
```

Out[10]: True

Applying the exterior derivative to the same form twice always yields 0 (i.e., the form identically equal to 0).

In [11]:
```
ddf = df.exterior_derivative();
print(ddf)
ddf.display()
```

```
2-form on the 3-dimensional differentiable manifold U
```

Out[11]: 0

In [12]:
```
ddt1 = dt1.exterior_derivative();
print(ddt1)
ddt1.display()
```

```
3-form on the 3-dimensional differentiable manifold U
```

Out[12]: $\mathrm{d}\mathrm{d}\theta_1 = 0$

In [13]:
```
ddt2 = dt2.exterior_derivative();
print(ddt2)
ddt2.display()
```

```
3-form on the 3-dimensional differentiable manifold U
```

Out[13]: $\mathrm{d}\mathrm{d}\theta_2 = 0$

Differential k-forms map $\mathfrak{X}^k \to \mathfrak{F}$.

In [14]:
```
X1 = U.vector_field(latex_name=r'X_1');
X1[:] = [x,-x,2*z];
X2 = U.vector_field(latex_name=r'X_2');
X2[:] = [cos(z),-sin(z),z];
```

In [15]:
```
print(df(X1))
df(X1).display()
```

Scalar field on the 3-dimensional differentiable manifold U

Out[15]:
$$\begin{array}{lcl} \mathbb{U} & \longrightarrow & \mathbb{R} \\ (x, y, z) & \longmapsto & 2\,xz^{\sin(y)}\cos(y)\log(z) + 2\,x^2 - 4\,z^{\sin(y)}\sin(y) \end{array}$$

$df(X_1)$ and $X_1(f)$ coincide. In Calc III lingo, they both represent the (non-normalized) directional derivative of f in the direction of X_1.

In [16]:
```
print(X1(f))
X1(f).display()
```

Scalar field on the 3-dimensional differentiable manifold U

Out[16]:
$$\begin{array}{lcl} \mathbb{U} & \longrightarrow & \mathbb{R} \\ (x, y, z) & \longmapsto & 2\,xz^{\sin(y)}\cos(y)\log(z) + 2\,x^2 - 4\,z^{\sin(y)}\sin(y) \end{array}$$

In [17]:
```
df(X1) == X1(f)
```

Out[17]: True

2-forms take two vector fields as arguments.

In [18]:
```
dt1(X1,X2).display()
```

Out[18]:
$$\begin{array}{lcl} \mathrm{d}\theta_1\,(X_1, X_2):\ \mathbb{U} & \longrightarrow & \mathbb{R} \\ (x, y, z) & \longmapsto & -(x\cos(y) - 2\,x)z - x\cos(z) + (2\,z(\cos \end{array}$$

In [19]:
```
dt1(X2,X1).display()
```

Out[19]:
$$\begin{array}{lcl} \mathrm{d}\theta_1\,(X_2, X_1):\ \mathbb{U} & \longrightarrow & \mathbb{R} \\ (x, y, z) & \longmapsto & (x\cos(y) - 2\,x)z + x\cos(z) - (2\,z(\cos(\end{array}$$

In [20]:
```
dt1(X1,X2) == -dt1(X2,X1)
```

Out[20]: True

The space Ω^n is a 1-dimensional space whose basis covector is the *Euclidean volume form* on $\mathbb{U}$.

In [21]:
```
vol = U.diff_form(3,latex_name=r'\omega');
vol[1,2,3] = 1;
```

In [22]:
```
print(vol)
```

```
3-form on the 3-dimensional differentiable manifold U
```

In [23]:
```
vol.display()
```

Out[23]: $\omega = \mathrm{d}x \wedge \mathrm{d}y \wedge \mathrm{d}z$

In [24]:
```
vol(2*e[1],4*e[2],e[3])
```

Out[24]: Scalar field on the 3-dimensional differentiable manifold U

In [25]:
```
vol(2*e[1],4*e[2],e[3]).display()
```

Out[25]:
$$\begin{array}{lcl} \mathbb{U} & \longrightarrow & \mathbb{R} \\ (x, y, z) & \longmapsto & 8 \end{array}$$

In [26]:
```
p=U.point((1,2,3))
```

In [27]:
```
vol(2*e[1],4*e[2],e[3])(p)
```

Out[27]: 8

The volume form can be used to calculate the determinant of a linear transformation of $T\mathbb{U}$. By a linear transformation of $T\mathbb{U}$, we mean a linear transformation of each tangent space that varies smoothly between tangent spaces. We call such a map a *tangent-space automorphism field* on $\mathbb{U}$.

In [28]:
```
A = U.automorphism_field('aut',latex_name=r'A');
A[e,:] = [[1,x,z],[0,3,y],[0,0,5]];
print(A)
A.display()
```

```
Field of tangent-space automorphisms aut on the 3-dimensional differentiable manifold U
```

Out[28]:
$$A = \frac{\partial}{\partial x} \otimes \mathrm{d}x + x\frac{\partial}{\partial x} \otimes \mathrm{d}y + z\frac{\partial}{\partial x} \otimes \mathrm{d}z + 3\frac{\partial}{\partial y} \otimes \mathrm{d}y + y\frac{\partial}{\partial y} \otimes \mathrm{d}z + 5\frac{\partial}{\partial z}$$

In [29]:
```
A.display_comp()
```

Out[29]:
$$\begin{aligned} A^{x}{}_{x} &= 1 \\ A^{x}{}_{y} &= x \\ A^{x}{}_{z} &= z \\ A^{y}{}_{y} &= 3 \\ A^{y}{}_{z} &= y \\ A^{z}{}_{z} &= 5 \end{aligned}$$

In [30]:
```
A[e,:]
```

Out[30]:
$$\begin{pmatrix} 1 & x & z \\ 0 & 3 & y \\ 0 & 0 & 5 \end{pmatrix}$$

In [31]:
```
detA = vol(A(e[1]),A(e[2]),A(e[3]));
detA.display()
```

Out[31]:
$$\omega\left(A\left(\frac{\partial}{\partial x}\right), A\left(\frac{\partial}{\partial y}\right), A\left(\frac{\partial}{\partial z}\right)\right): \begin{array}{ccc} \mathbb{U} & \longrightarrow & \mathbb{R} \\ (x, y, z) & \longmapsto & 15 \end{array}$$

Of course, in general the determinant depends on the point $p \in \mathbb{U}$.

In [32]:
```
B = U.automorphism_field('B',latex_name=r'B');
B[e,:] = [[x,0,0],[0,y,0],[0,0,exp(z)]];
B.display()
```

Out[32]:
$$B = x\frac{\partial}{\partial x} \otimes \mathrm{d}x + y\frac{\partial}{\partial y} \otimes \mathrm{d}y + e^{z}\frac{\partial}{\partial z} \otimes \mathrm{d}z$$

In [33]:
```
B.display_comp()
```

Out[33]:
$$B^{x}{}_{x} = x$$
$$B^{y}{}_{y} = y$$
$$B^{z}{}_{z} = e^{z}$$

In [34]:
```
B[e,:]
```

Out[34]:
$$\begin{pmatrix} x & 0 & 0 \\ 0 & y & 0 \\ 0 & 0 & e^{z} \end{pmatrix}$$

In [35]:
```
detB = vol(B(e[1]),B(e[2]),B(e[3]));
detB.display()
```

Out[35]:
$$\omega\left(B\left(\frac{\partial}{\partial x}\right), B\left(\frac{\partial}{\partial y}\right), B\left(\frac{\partial}{\partial z}\right)\right): \mathbb{U} \longrightarrow \mathbb{R}$$
$$(x, y, z) \longmapsto xye^{z}$$

In [36]:
```
p = U.point((1,1,1),chart=coord);
detB(p)
```

Out[36]:
$$e$$

In [37]:
```
q = U.point((-1,1,0),chart=coord);
detB(q)
```

Out[37]:
$$-1$$

Finally, we discuss how to code the results of Exercise 1.6.9 in O'Neill's *Elementary Differential Geometry* book. For this we need a metric (think: Riesz Representation Theorem). We'll use the Euclidean metric, *a.k.a.* the dot product.

In [38]:
```
g = U.riemannian_metric('g',latex_name=r'g');
g[1,1] = g[2,2] = g[3,3] =1;
print(g)
g.display()
```

```
Riemannian metric g on the 3-dimensional differentiable manifold U
```

Out[38]:
$$g = \mathrm{d}x \otimes \mathrm{d}x + \mathrm{d}y \otimes \mathrm{d}y + \mathrm{d}z \otimes \mathrm{d}z$$

In [39]:
```
g.display_comp()
```

Out[39]:
$$\begin{aligned} g_{x\,x} &= 1 \\ g_{y\,y} &= 1 \\ g_{z\,z} &= 1 \end{aligned}$$

The gradient of a function f is the vector field dual to the differential df. Changing a covector field to a vector field is known as *raising the indices*, so the Sage command is `.up()`.

In [40]:
```
gradf = df.up(g);
gradf.display()
```

Out[40]:
$$2\,x\frac{\partial}{\partial x} - 2\,z^{\sin(y)}\cos(y)\log(z)\frac{\partial}{\partial y} + \left(-2\,z^{\sin(y)-1}\sin(y)\right)\frac{\partial}{\partial z}$$

In [41]:
```
gradf.display_comp()
```

Out[41]:
$$\begin{aligned} X^{x} &= 2\,x \\ X^{y} &= -2\,z^{\sin(y)}\cos(y)\log(z) \\ X^{z} &= -2\,z^{\sin(y)-1}\sin(y) \end{aligned}$$

In [42]:
```
df.display_comp()
```

Out[42]:
$$\begin{aligned} X_{x} &= 2\,x \\ X_{y} &= -2\,z^{\sin(y)}\cos(y)\log(z) \\ X_{z} &= -2\,z^{\sin(y)-1}\sin(y) \end{aligned}$$

The *curl* of a vector field X is the exterior derivative of the 1-form dual to X.

In [43]:
```
phi_X1 = X1.down(g);
```

To save key strokes, we can import a shortcut for `.exterior_derivative()` .

In [44]:
```
from sage.manifolds.utilities import xder
```

In [45]:
```
d_phi_X1 = xder(phi_X1);
print(d_phi_X1)
d_phi_X1.display()
```

```
2-form on the 3-dimensional differentiable manifold U
```

Out[45]: $-\mathrm{d}x \wedge \mathrm{d}y$

In [46]:
```
d_phi_X1 == phi_X1.exterior_derivative()
```

Out[46]: True

In terms of the basis of $\mathfrak{X}$, the curl of X_1 is:

In [47]:
```
d_phi_X1_dual= d_phi_X1.up(g);
d_phi_X1_dual.display()
```

Out[47]: $-\frac{\partial}{\partial x} \otimes \frac{\partial}{\partial y} + \frac{\partial}{\partial y} \otimes \frac{\partial}{\partial x}$

In [48]:
```
curl_X1 = U.vector_field(latex_name=r'\mathrm{curl}(X_1)');
curl_X1[:] = [0,0,d_phi_X1[1,2]];
curl_X1.display()
```

Out[48]: $\mathrm{curl}(X_1) = -\frac{\partial}{\partial z}$

Recall X_1 and verify that this corresponds with the result you'd obtain from Calc III.

In [49]:
```
X1.display()
```

Out[49]: $X_1 = x\frac{\partial}{\partial x} - x\frac{\partial}{\partial y} + 2\,z\frac{\partial}{\partial z}$

Finally, the divergence of a vector field X is the coefficient of the 3-form $d(\eta_X) = (\mathrm{div}X)\, dx \wedge dy \wedge dz$ Here, η_X is the 2-form corresponding to X as described in O'Neill's exercise.

In [50]:
```
eta_X1 = U.diff_form(2,latex_name=r'\eta_{X_1}');
eta_X1[1,2] = X1[3];
eta_X1[1,3] = -X1[2];
eta_X1[2,3] = X1[1];
eta_X1.display()
```

Out[50]: $\eta_{X_1} = 2\,z\mathrm{d}x \wedge \mathrm{d}y + x\mathrm{d}x \wedge \mathrm{d}z + x\mathrm{d}y \wedge \mathrm{d}z$

In [51]:
```
d_eta_X1 = xder(eta_X1);
d_eta_X1.display()
```

Out[51]: $\mathrm{d}\eta_{X_1} = 3\mathrm{d}x \wedge \mathrm{d}y \wedge \mathrm{d}z$

In [52]:
```
div_X1 = d_eta_X1[1,2,3];
div_X1.display()
```

Out[52]: $(x, y, z) \mapsto 3$

In [53]:
```
eta_X2 = U.diff_form(2,latex_name=r'\eta_{X_2}');
eta_X2[1,2] = X2[3];
eta_X2[1,3] = -X2[2];
eta_X2[2,3] = X2[1];
eta_X2.display()
```

Out[53]: $\eta_{X_2} = z\mathrm{d}x \wedge \mathrm{d}y + \sin(z)\mathrm{d}x \wedge \mathrm{d}z + \cos(z)\mathrm{d}y \wedge \mathrm{d}z$

In [54]:
```
d_eta_X2 = xder(eta_X2);
d_eta_X2.display()
```

Out[54]: $\mathrm{d}\eta_{X_2} = \mathrm{d}x \wedge \mathrm{d}y \wedge \mathrm{d}z$

In [55]:
```
div_X2 = d_eta_X2[1,2,3];
div_X2.display()
```

Out[55]: $(x, y, z) \mapsto 1$

The *Laplacian* of a function $f \in \mathfrak{F}$ is $\Delta f = \mathrm{div}(\mathrm{grad}\, f)$.

In [56]:
```
eta_f = U.diff_form(2,latex_name=r'\eta_{f}');
eta_f[1,2] = gradf[3];
eta_f[1,3] = -gradf[2];
eta_f[2,3] = gradf[1];
eta_f.display()
```

Out[56]: $\eta_f = \left(-2\,z^{\sin(y)-1}\,\sin(y)\right)\mathrm{d}x \wedge \mathrm{d}y + 2\,z^{\sin(y)}\,\cos(y)\,\log(z)\mathrm{d}x \wedge \mathrm{d}z + 2\,x\mathrm{d}y$

In [57]:
```
Laplacian_f = xder(eta_f)[1,2,3];
```

In [58]:
```
Laplacian_f.display()
```

Out[58]:
$$(x, y, z) \mapsto -\frac{2\left(z^{\sin(y)+2}\cos(y)^2\log(z)^2 - z^{\sin(y)+2}\log(z)\sin(y) + \left(\sin(y)\right.\right.}{z^2}$$

In [ ]: