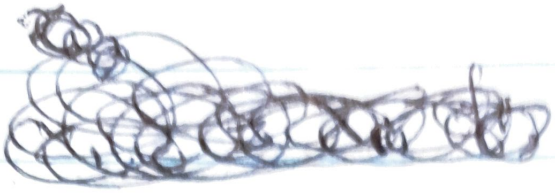
1.) Assignment - 12

a).



$$x_{i,c} = \lambda_{c1} f_{i,v} + \lambda_{c2} f_{i,p} + \lambda_{c3} f_{i,m} + \varepsilon_i + \mu_c$$

$$x_{i,s} = \lambda_{s1} f_{i,v} + \lambda_{s2} f_{i,p} + \lambda_{s3} f_{i,m} + \varepsilon_i + \mu_s$$

$$x_{i,o} = \lambda_{o1} f_{i,v} + \lambda_{o2} f_{i,p} + \lambda_{o3} f_{i,m} + \varepsilon_i + \mu_o$$

$$x_{i,n} = \lambda_{n1} f_{i,v} + \lambda_{n2} f_{i,p} + \lambda_{n3} f_{i,m} + \varepsilon_i + \mu_n$$

$$\begin{bmatrix} c & o & s & n \\ x_{c1} & x_{01} & x_{s1} & x_{n1} \\ x_{c2} & x_{02} & x_{s2} & x_{n2} \\ \vdots & & & \\ x_{c50} & x_{050} & x_{s50} & x_{n50} \end{bmatrix} \times \underset{50\times4}{} = \underset{50\times3}{\begin{bmatrix} f_{1,v} & f_{1,p} & f_{1,m} \\ f_{2,v} & f_{2,p} & f_{2,m} \\ \vdots & \vdots & \vdots \\ f_{50,v} & f_{50,p} & f_{50,m} \end{bmatrix}} \underset{3\times4}{\begin{bmatrix} \lambda_{c1} & \lambda_{01} & \lambda_{s1} & \lambda_{n1} \\ \lambda_{c2} & \lambda_{01} & \lambda_{s2} & \lambda_{n2} \\ \lambda_{c3} & \lambda_{02} & \lambda_{s3} & \lambda_{n3} \end{bmatrix}}$$

$$\underset{50\times4}{\begin{bmatrix} \mu_c & \mu_0 & \mu_s & \mu_n \\ \mu_c & \mu_0 & \mu_s & \mu_n \\ \vdots & & & \\ \mu_c & \mu_0 & \mu_s & \mu_n \end{bmatrix}} + \underset{50\times4}{\begin{bmatrix} \varepsilon_{1,1} & \varepsilon_{1,2} & \varepsilon_{1,3} & \varepsilon_{1,4} \\ \varepsilon_{2,1} & \varepsilon_{2,2} & \varepsilon_{2,3} & \varepsilon_{2,4} \\ \vdots & & & \\ \varepsilon_{50,1} & \varepsilon_{50,2} & \varepsilon_{50,3} & \varepsilon_{50,4} \end{bmatrix}}$$

$$X = f\Lambda + \varepsilon + \mu$$

$$\underset{(50\times4)}{} \quad \underbrace{\underset{(50\times3)(3\times4)}{}}_{50\times4} \quad \underset{(50\times4)}{} \cdot \underset{(50\times4)}{}$$

c).

Known term = X

To be estimated : Λ, f, ε.

d).

(i) $f \sim N(0, I)$.

(ii) $x | f \sim N(\mu + \Lambda f, \Psi)$.

e)

In [14]:
```
using Distributions;
using Gadfly;
```

In [3]:
```
mu = [10 20 30 40]';
Lambda =
[1.0 0 0
0 1.0 0
0 0 1.0
0.5 0.5 0];
Psi = diagm([0.1, 0.2, 0.3, 0.4]);
d1 = MvNormal([0,0,0],ones(3));
X = zeros(50,4);
for i=1:50
    f = rand(d1,1);
    d2 = MvNormal(vec(mu+ Lambda*f),Psi);
    x = rand(d2,1);
    X[i,:] = x';
end
```

In [4]:
```
function E_Step(X,mu,Lambda,Psi,k)
    mu_f_by_x = (X - repmat(mu',size(X,1),1))*(Lambda'*inv(Lambda*Lambda' + Ps
i))';
    Sig_f_by_x = eye(k) - Lambda'*inv(Lambda*Lambda' + Psi)*Lambda;
    return mu_f_by_x,Sig_f_by_x;
end
```

Out[4]: E_Step (generic function with 1 method)

In [6]:
```
function M_Step(X,mu_f_by_x,Sig_f_by_x,k)
    nrows, ncols = size(X);
    #Computing mu
    mu = mean(X,1)';
    #Computing Lambda
    Lambda_term1 = zeros(ncols,k);
    Lambda_term2 = zeros(k,k);
    for i=1:nrows
        Lambda_term1 = Lambda_term1 + ((X[i,:] - mu)*mu_f_by_x[i,:]');
        Lambda_term2 = Lambda_term2 + (mu_f_by_x[i,:]*mu_f_by_x[i,:]')+Sig_f_b
y_x;
    end
    Lambda = Lambda_term1*inv(Lambda_term2);
    #Computing Psi
    Phi = zeros(ncols,ncols);
    for i=1:nrows
        Phi = Phi + (X[i,:]*X[i,:]' - X[i,:]*mu_f_by_x[i,:]'*Lambda' -
            Lambda*mu_f_by_x[i,:]*X[i,:]' +
            Lambda*(mu_f_by_x[1,:]*mu_f_by_x[1,:]'+Sig_f_by_x)*Lambda')
    end
    Psi = diagm(diag(Phi./nrows));
    return mu, Lambda, Psi
end
```

Out[6]: M_Step (generic function with 1 method)

In [8]:
```
function compute_llh(X,mu,Lambda,Psi)
    llh = 0;
    for i=1:size(X,1)
        llh = llh + log(pdf(MvNormal(vec(mu),(Lambda*Lambda')+Psi),X[i,:]));
    end
return llh;
end
```

Out[8]: compute_llh (generic function with 1 method)

In [ ]:
```
function fa_em(X,k)
    max_Iter = 100;
    eps = 0.0001;
    llh = -Inf*ones(max_Iter+1);
    mu = mean(X,1)';
    Lambda = rand(size(X,2),k);
    Psi = diagm(rand(size(X,2)));
    #print(mu,"\n",Lambda,"\n",Psi,"\n");
    llh[1] = compute_llh(X,mu,Lambda,Psi);
    #print(llh[1],"\n")
    for i=1:max_Iter
        #print(i,"\n");
        mu_f_by_x,Sig_f_by_x = E_Step(X,mu,Lambda,Psi,k);
        mu_new, Lambda_new, Psi_new = M_Step(X,mu_f_by_x,Sig_f_by_x,k);
        #print(mu_new,"\n",Lambda_new,"\n",Psi_new,"\n");
        llh[i+1] = compute_llh(X,mu_new,Lambda_new,Psi_new);
        #print(llh[i+1],"\n");
        if(sum(abs.(mu_new-mu))<eps && sum(abs.(Lambda_new-Lambda))<eps && sum
(abs.(Psi_new-Psi))<eps)
            break;
        end
        mu = mu_new;
        Lambda = Lambda_new;
        Psi = Psi_new;
        end
    mu_f_by_x,Sig_f_by_x = E_Step(X,mu,Lambda,Psi,k);
    return mu, Lambda, Psi, mu_f_by_x, Sig_f_by_x, llh;
end
```

f) Mu is converging but lambda and psi are not converging

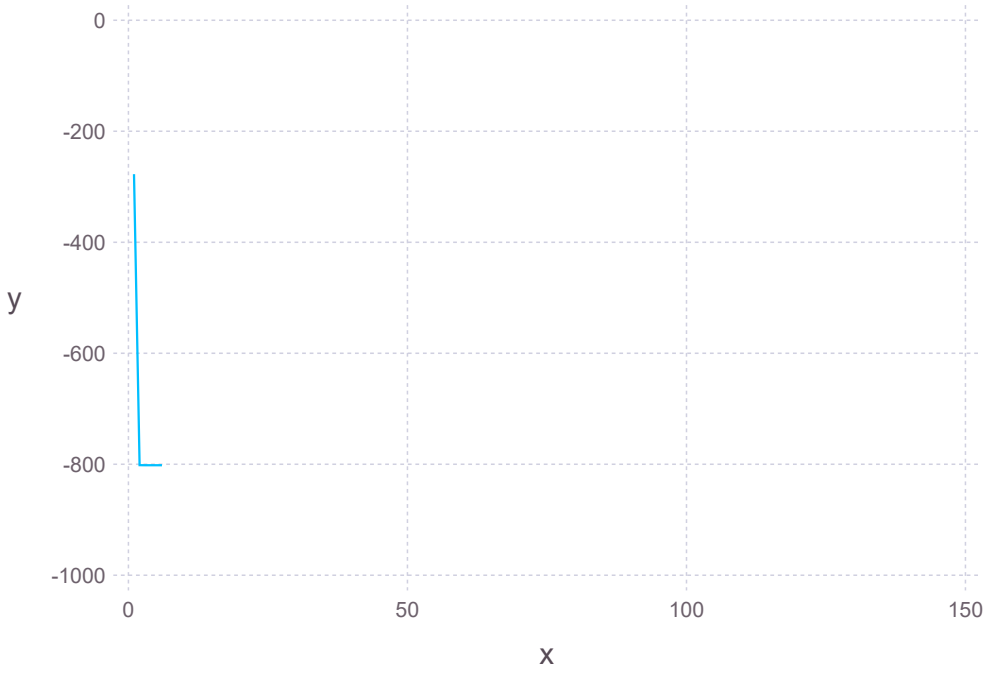
In [13]:
```
mu, Lambda, Psi, mu_f_by_x, Sig_f_by_x, llh = fa_em(X,3)
```

Out[13]:
```
([9.77027; 19.8353; 29.6944; 39.9425], [1.88283e-7 1.0449e-7 2.75795e-7; 6.62
089e-8 3.76831e-8 9.09447e-8; 6.92704e-8 3.95987e-8 1.00074e-7; 1.20567e-7 6.
72345e-8 1.74176e-7], [96.2455 0.0 0.0 0.0; 0.0 394.619 0.0 0.0; 0.0 0.0 882.
899 0.0; 0.0 0.0 0.0 1596.21], [-2.01034e-9 -1.11577e-9 -2.94751e-9; 1.32569e
-9 7.34241e-10 1.94987e-9; … ; 8.75746e-10 4.86741e-10 1.27388e-9; 7.67053e-1
1 4.16959e-11 1.1935e-10], [1.0 -2.18919e-16 -5.75798e-16; -2.18919e-16 1.0 -
3.19928e-16; -5.75798e-16 -3.19928e-16 1.0], [-277.39, -801.652, -801.697, -8
01.697, -801.697, -801.697, -Inf, -Inf, -Inf, -Inf  …  -Inf, -Inf, -Inf, -In
f, -Inf, -Inf, -Inf, -Inf, -Inf, -Inf])
```

g) Yes it is converging to -inf after x = 10 or so

In [15]:
```
plot(x=collect(1:1:101), y=llh,Geom.line)
```

Out[15]:



h) Again only mu is converging but psi & lambda are not getting coverged.

Mu is same for k=2 & k=3 Lambda is not same for k=2 & k=3 Psi is same for k=2 & k=3

In [16]:
```
mu_2, Lambda_2, Psi_2, mu_f_by_x_2, Sig_f_by_x_2, llh_2 = fa_em(X,2)
```

Out[16]:
```
([9.77027; 19.8353; 29.6944; 39.9425], [3.04037e-7 7.68734e-8; 1.01739e-7 2.6
0868e-8; 1.10496e-7 2.88594e-8; 1.92618e-7 4.88022e-8], [96.2455 0.0 0.0 0.0;
0.0 394.619 0.0 0.0; 0.0 0.0 882.899 0.0; 0.0 0.0 0.0 1596.21], [-3.24857e-9
-8.2169e-10; 2.14762e-9 5.42334e-10; … ; 1.40664e-9 3.55616e-10; 1.29817e-10
3.25958e-11], [1.0 -2.59068e-16; -2.59068e-16 1.0], [-366.288, -801.593, -80
1.697, -801.697, -801.697, -801.697, -Inf, -Inf, -Inf, -Inf  …  -Inf, -Inf, -
Inf, -Inf, -Inf, -Inf, -Inf, -Inf, -Inf, -Inf])
```

g) Yes it is converging to -inf after x = 10 or so

In [17]:
```
plot(x=collect(1:1:101), y=llh_2,Geom.line)
```

Out[17]: