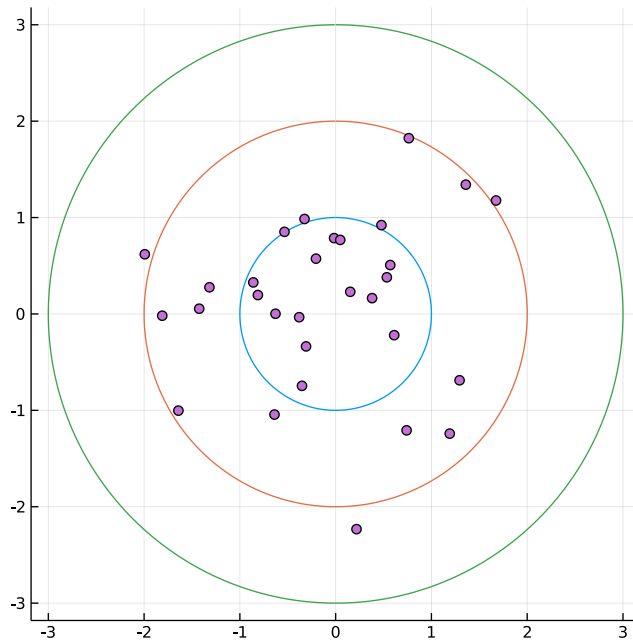```
using LinearAlgebra, Plots
gr(size=(500, 500), legend=false)
```

```
Plots.GRBackend()
```

# Principal Component Analysis

An *affine transformation* converts a standard normal distribution into a possibly non-centered and correlated normal distribution.

```
xx = 0:0.01:2π
d = randn(30, 2)
plot(sin.(xx), cos.(xx))
plot!(2 .* sin.(xx), 2 .* cos.(xx))
plot!(3 .* sin.(xx), 3 .* cos.(xx))
scatter!(d[:,1], d[:,2])
```



```
transformation(x) = (x * [1. 3; 2 1]  .+ [1. 2])
x1 = transformation([sin.(xx) cos.(xx)])
x2 = transformation(2 .*[sin.(xx) cos.(xx)])
x3 = transformation(3 .*[sin.(xx) cos.(xx)])
plot(x1[:,1], x1[:,2])
plot!(x2[:,1], x2[:,2])
plot!(x3[:,1], x3[:,2])
dt = transformation(d)
scatter!(dt[:,1], dt[:,2])
```
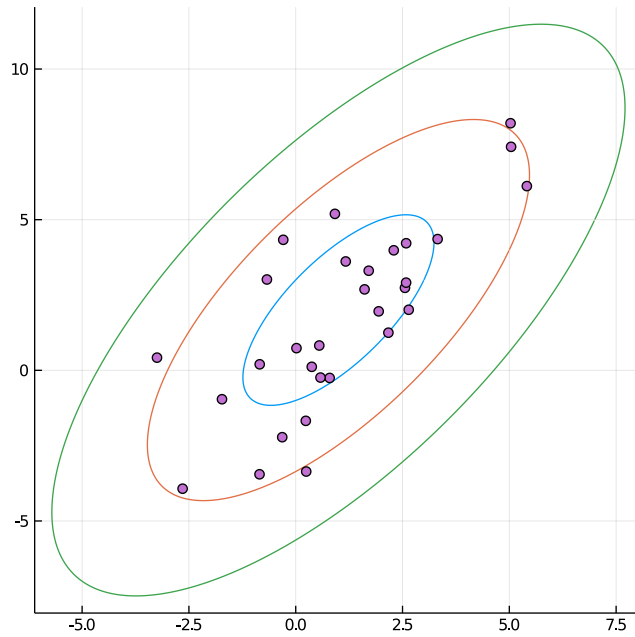
First goals of PCA:

- identify original center
- identify main axes of ellipsoid (eigenvectors)

This boils down to the task of tranforming the data back into standard normally distributed ones.

Final goal of PCA:

- reduce dimensionality of data by preserving as much information as possible

PCA relies on the assumption that data points a **independent** draws from a transformed normal distribution.

Population data (including language data) are not independent.

Illustration: Chris' data:
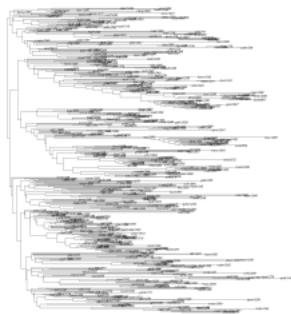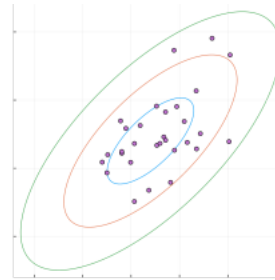
	C_H	C_WALS	C_TTR	C_D	C_A
	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
mapu1245	-0.833590662	0.5031136	0.08595837	0.30223334	-0.09905640
kore1280	2.037912427	0.4595238	0.14744375	0.44032458	0.06643120
chip1262	0.742597976	0.6666667	0.11049524	0.40909720	-0.06590704
coat1238	-0.120044872	0.4000000	0.09030347	0.29307938	-0.13468380
tlah1239	-0.114511276	1.0000000	0.12513113	0.36829698	-0.06784751
high1276	-0.314481772	0.8333333	0.06551463	0.22475648	-0.11082072
xico1235	-0.007917082	0.4062500	0.09455093	0.29188991	0.01392109
hueh1236	-0.826923489	0.5555556	0.07449885	0.22925591	-0.09627536

Before we can apply PCA to the columns, we have to **decorrelate** the rows!

Phylogenies as matrices

Let us assume a phylogeny $\mathcal{T}$ with n branches and m tips, and a character x evolving along $\mathcal{T}$ according to a Brownian motion. Evolution along each branch i ($1 \leq i \leq n$) is drawn from an independent normal distribution with mean 0 and variance l_i, leading to a vector B of length n which is multivariate normal with mean $\mathbf{0}$ and covariance matrix $\operatorname{diag}(l_1 \cdots l_n) \doteq \Lambda$.

The state of x at the root is x_r.

The topology of $\mathcal{T}$ can be represented by an $n \times n$ matrix T where

$$t_{ij} = \begin{cases} 1 \text{ if branch } j \text{ is on the path from} \\ \quad \text{the root to the node at the bottom of branch } j \\ 0 \text{ else} \end{cases}$$

(Alternatively, one can identify the rows and columns with the non-root nodes, where $t_{ij} = 1$ means that j is on the path from the root to i, including i itself. So T is essentially the transitive closure of the adjacency matrix of $\mathcal{T}$.)
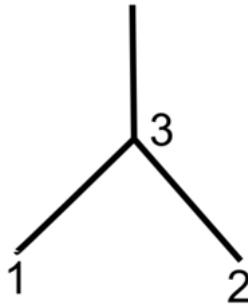
Let us consider a few examples.

```
using SymPy

sy(A) = convert(Array{Sym, 2}, A)

function sySVD(A)
    V, D = (A.T * A).diagonalize(normalize=true, sort=true);
    S = reverse(sqrt.(D[D.> 0]))
    V = V[:,reverse(1:size(V,2))]
    V = V[:,1:size(A,1)]
    U = A * V * Diagonal(1 ./ S)
    return (U, S, V)
end;

function syPinv(A)
    U, S, V = sySVD(A)
    return V * Diagonal(1 ./ S) * U.T
end;
```
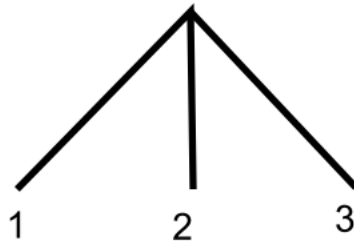


```
T = sy([
    1 0 1
    0 1 1
    0 0 1
])
```

$$\begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix}$$

```
T = sy([
    1 0 0
    0 1 0
    0 0 1
])
```

$$
\begin{bmatrix}
1 & 0 & 0 \\
0 & 1 & 0 \\
0 & 0 & 1
\end{bmatrix}
$$


```
T = sy([
    1 0 0 0 1 0
    0 1 0 0 1 0
    0 0 1 0 0 1
    0 0 0 1 0 1
    0 0 0 0 1 0
    0 0 0 0 0 1
])
```

$$
\begin{bmatrix}
1 & 0 & 0 & 0 & 1 & 0 \\
0 & 1 & 0 & 0 & 1 & 0 \\
0 & 0 & 1 & 0 & 0 & 1 \\
0 & 0 & 0 & 1 & 0 & 1 \\
0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 1
\end{bmatrix}
$$

Let b be a vector of branch-wise Brownian motion steps, i.e., a random vector drawn from

$$b \sim \mathcal{N}(\mathbf{0}, \Lambda)$$

Then $x = Tb + x_r \mathbf{1}$ is the vector of values of x at the nodes (including the internal ones). x is distributed according to a multivarate normal distribution

$$x \sim \mathcal{N}(x_r \mathbf{1}, \Sigma),$$

where $\sigma_{ij} = \sum_k t_{ik} t_{il} t l_k$. Proof:

$$\begin{aligned}
Cov((Tb + x_r\mathbf{1})_i, (Tb + x_r\mathbf{1})_j) &= E((Tb + x_r\mathbf{1})_i(Tb + x_r\mathbf{1})_j) - E((Tb + x_r\mathbf{1})_i)E((Tb + x_r\mathbf{1})_j) \\
&= E(((Tb)_i + x_r)((Tb)_j + x_r)) - E((Tb)_i + x_r)E((Tb)_j + x_r) \\
&= E((Tb)_i(Tb)_j + ((Tb)_i + (Tb)_j)x_r + x_r^2) - x_r^2 \\
&= E((Tb)_i(Tb)_j) + x_r E((Tb)_i + (Tb)_j) \\
&= E((Tb)_i(Tb)_j) \\
&= E(\sum_k t_{ik}b_k \sum_l t_{jl}b_l) \\
&= \sum_{kl} t_{ik}a_{jl}E(b_k b_l) \\
&= \sum_k t_{ik}t_{jk}l_k
\end{aligned}$$

It is easy to see that

$$\begin{aligned}
\Sigma &= T\Lambda T' \\
&= (T\Lambda^{\frac{1}{2}})(T\Lambda^{\frac{1}{2}})'
\end{aligned}$$

Let x_t be the restriction of x to the tips, and T_t the restriction of T to the tip rows. Then by the same token,
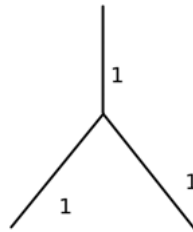
$$x_t \sim \mathcal{N}(x_r\mathbf{1}, (T_t\Lambda^{\frac{1}{2}})(T_t\Lambda^{\frac{1}{2}})')$$

For simplicity's sake, I define

$$\Phi \doteq T_t\Lambda^{\frac{1}{2}}$$

Φ is the result of multiplying each column of the topology matrix T with the square root of the corresponding branch length and restricting the rows to the tips.

Examples:


```
Φ = sy([
    1 0 1
    0 1 1
])
```

$$\begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix}$$
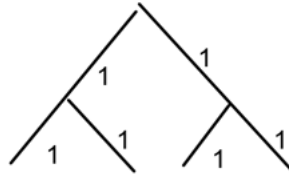


```
Φ = sy([
    1 0 0
    0 1 0
    0 0 1
])
```

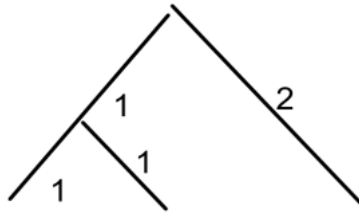$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

```
Φ = sy([
    1 0 0 0 1 0
    0 1 0 0 1 0
    0 0 1 0 0 1
    0 0 0 1 0 1
])
```

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 \end{bmatrix}$$


```
Φ = sy([
    1 0 0         1
    0 1 0         1
    0 0 sympy.sqrt(2) 0
])
```

$$\begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & \sqrt{2} & 0 \end{bmatrix}$$

$\Phi\Phi'$ is the covariance matrix of x_t. It is easy to see that $(\Phi\Phi')_{ij}$ is the length of the path from the root to the latest common ancestor of tips i and j.

```
Φ * Φ.T
```

$$\begin{bmatrix} 2 & 1 & 0 \\ 1 & 2 & 0 \\ 0 & 0 & 2 \end{bmatrix}$$

Φ can be seen as a linear transformation that maps a standard normally distributed vector to $x - x_r$. The **Singular Value Decomposition** of Φ decomposes this transformation into three components:
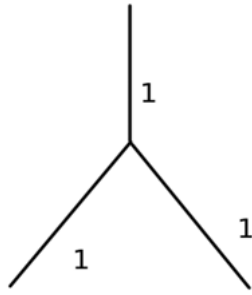
- V^T changes the coordinate system into Φ's eigenspace,
- S stretches or shrinks along the dimensions of the eigenspace, and
- U changes the coordinates again, mapping into the system of the standard normally distributed random variable.

It holds that

$$\Phi = USV^T,$$

and this factorization is always possible.

Examples

```
Φ = sy([
    1 0 1
    0 1 1
])
```

$$\begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix}$$

```
U,S,V = sySVD(Φ);
```

```
U
```

$$\begin{bmatrix} \frac{\sqrt{2}}{2} & -\frac{\sqrt{2}}{2} \\ \frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} \end{bmatrix}$$

```
Diagonal(S)
```

$$\begin{bmatrix} \sqrt{3} & 0 \\ 0 & 1 \end{bmatrix}$$

```
V
```

$$\begin{bmatrix} \frac{\sqrt{6}}{6} & -\frac{\sqrt{2}}{2} \\ \frac{\sqrt{6}}{6} & \frac{\sqrt{2}}{2} \\ \frac{\sqrt{6}}{3} & 0 \end{bmatrix}$$


```
Φ = sy([
    1 0 0
    0 1 0
    0 0 1
])
```

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

```
U, S, V = sySVD(Φ);
```
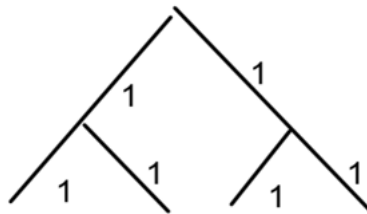
```
U
```

$$\begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix}$$

```
diagm(S)
```

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

```
V
```

$$\begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix}$$


```
Φ = sy([
    1 0 0 0 1 0
    0 1 0 0 1 0
    0 0 1 0 0 1
    0 0 0 1 0 1
])
```

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 \end{bmatrix}$$
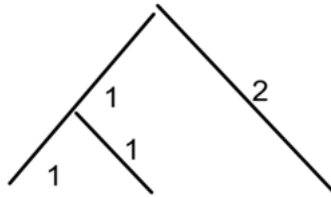
```
U, S, V = sySVD(Φ);
```

```
U
```

$$\begin{bmatrix} 0 & \frac{\sqrt{2}}{2} & 0 & -\frac{\sqrt{2}}{2} \\ 0 & \frac{\sqrt{2}}{2} & 0 & \frac{\sqrt{2}}{2} \\ \frac{\sqrt{2}}{2} & 0 & -\frac{\sqrt{2}}{2} & 0 \\ \frac{\sqrt{2}}{2} & 0 & \frac{\sqrt{2}}{2} & 0 \end{bmatrix}$$

```
diagm(S)
```

$$\begin{bmatrix} \sqrt{3} & 0 & 0 & 0 \\ 0 & \sqrt{3} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

```
V
```

$$\begin{bmatrix} 0 & \frac{\sqrt{6}}{6} & 0 & -\frac{\sqrt{2}}{2} \\ 0 & \frac{\sqrt{6}}{6} & 0 & \frac{\sqrt{2}}{2} \\ \frac{\sqrt{6}}{6} & 0 & -\frac{\sqrt{2}}{2} & 0 \\ \frac{\sqrt{6}}{6} & 0 & \frac{\sqrt{2}}{2} & 0 \\ 0 & \frac{\sqrt{6}}{3} & 0 & 0 \\ \frac{\sqrt{6}}{3} & 0 & 0 & 0 \end{bmatrix}$$


```
Φ = sy([
    1 0 0       1
    0 1 0       1
    0 0 sympy.sqrt(2) 0
])
```

$$\begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & \sqrt{2} & 0 \end{bmatrix}$$
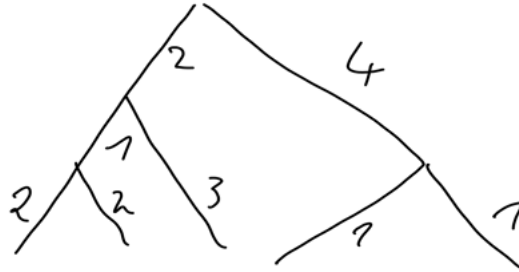
```
U, S, V = sySVD(Φ);
```

```
U
```

$$\begin{bmatrix} \frac{\sqrt{2}}{2} & 0 & -\frac{\sqrt{2}}{2} \\ \frac{\sqrt{2}}{2} & 0 & \frac{\sqrt{2}}{2} \\ 0 & 1 & 0 \end{bmatrix}$$

```
Diagonal(S)
```

$$\begin{bmatrix} \sqrt{3} & 0 & 0 \\ 0 & \sqrt{2} & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

```
V
```

$$\begin{bmatrix} \frac{\sqrt{6}}{6} & 0 & -\frac{\sqrt{2}}{2} \\ \frac{\sqrt{6}}{6} & 0 & \frac{\sqrt{2}}{2} \\ 0 & 1 & 0 \\ \frac{\sqrt{6}}{3} & 0 & 0 \end{bmatrix}$$


```
Φ = sy([
        sympy.sqrt(2) 0 0 0 0 1 sympy.sqrt(2) 0
        0 sympy.sqrt(2) 0 0 0 1 sympy.sqrt(2) 0
        0 0 sympy.sqrt(3) 0 0 0 sympy.sqrt(2) 0
        0 0 0 1 0 0 0 2
        0 0 0 0 1 0 0 2
        ])
```

$$\begin{bmatrix} \sqrt{2} & 0 & 0 & 0 & 0 & 1 & \sqrt{2} & 0 \\ 0 & \sqrt{2} & 0 & 0 & 0 & 1 & \sqrt{2} & 0 \\ 0 & 0 & \sqrt{3} & 0 & 0 & 0 & \sqrt{2} & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 2 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 2 \end{bmatrix}$$

```
Φ * Φ.T
```

$$\begin{bmatrix} 5 & 3 & 2 & 0 & 0 \\ 3 & 5 & 2 & 0 & 0 \\ 2 & 2 & 5 & 0 & 0 \\ 0 & 0 & 0 & 5 & 4 \\ 0 & 0 & 0 & 4 & 5 \end{bmatrix}$$

```
U, S, V = sySVD(Φ);
```

```
U
```

$$\begin{bmatrix} \frac{-\frac{5\sqrt{2}}{4}+\frac{\sqrt{82}}{4}}{\sqrt{\left(-\frac{\sqrt{246}}{4}+\frac{7\sqrt{6}}{4}\right)^2+\left(-\frac{5\sqrt{2}}{4}+\frac{\sqrt{82}}{4}\right)^2+2\left(-\frac{5}{4}+\frac{\sqrt{41}}{4}\right)^2+1}}+\frac{\sqrt{2}\left(-\frac{5}{4}+\frac{\sqrt{41}}{4}\right)}{\sqrt{\left(-\frac{\sqrt{246}}{4}+\frac{7\sqrt{6}}{4}\right)^2+\left(-\frac{5\sqrt{2}}{4}+\frac{\sqrt{82}}{4}\right)^2+2\left(-\frac{5}{4}+\frac{\sqrt{41}}{4}\right)^2+1}}+\frac{\sqrt{2}}{\sqrt{\left(-\frac{\sqrt{246}}{4}+\frac{7\sqrt{6}}{4}\right)^2+\left(-\frac{5\sqrt{2}}{4}+\frac{\sqrt{82}}{4}\right)^2+2\left(-\frac{5}{4}+\frac{\sqrt{41}}{4}\right)^2+1}} & \cdots \\ \cdots & \cdots \\ \cdots & \cdots \\ 0 & \\ 0 & \end{bmatrix}$$

S

$$\begin{bmatrix} \sqrt{\dfrac{\sqrt{41}}{2} + \dfrac{13}{2}} \\ \sqrt{\dfrac{13}{2} - \dfrac{\sqrt{41}}{2}} \\ 3 \\ \sqrt{2} \\ 1 \end{bmatrix}$$

V

$$\begin{bmatrix} \dfrac{-\frac{5}{4}+\frac{\sqrt{41}}{4}}{\sqrt{\left(-\frac{\sqrt{246}}{4}+\frac{7\sqrt{6}}{4}\right)^2+\left(-\frac{5\sqrt{2}}{4}+\frac{\sqrt{82}}{4}\right)^2+2\left(-\frac{5}{4}+\frac{\sqrt{41}}{4}\right)^2+1}} & \dfrac{-\frac{\sqrt{41}}{4}-\frac{5}{4}}{\sqrt{1+\left(\frac{5\sqrt{2}}{4}+\frac{\sqrt{82}}{4}\right)^2+2\left(\frac{5}{4}+\frac{\sqrt{41}}{4}\right)^2+\left(\frac{\sqrt{246}}{4}+\frac{7\sqrt{6}}{4}\right)^2}} & 0 & -\dfrac{\sqrt{2}}{2} & 0 \\[4mm]
\dfrac{-\frac{5}{4}+\frac{\sqrt{41}}{4}}{\sqrt{\left(-\frac{\sqrt{246}}{4}+\frac{7\sqrt{6}}{4}\right)^2+\left(-\frac{5\sqrt{2}}{4}+\frac{\sqrt{82}}{4}\right)^2+2\left(-\frac{5}{4}+\frac{\sqrt{41}}{4}\right)^2+1}} & \dfrac{-\frac{\sqrt{41}}{4}-\frac{5}{4}}{\sqrt{1+\left(\frac{5\sqrt{2}}{4}+\frac{\sqrt{82}}{4}\right)^2+2\left(\frac{5}{4}+\frac{\sqrt{41}}{4}\right)^2+\left(\frac{\sqrt{246}}{4}+\frac{7\sqrt{6}}{4}\right)^2}} & 0 & \dfrac{\sqrt{2}}{2} & 0 \\[4mm]
\dfrac{-\frac{\sqrt{246}}{4}+\frac{7\sqrt{6}}{4}}{\sqrt{\left(-\frac{\sqrt{246}}{4}+\frac{7\sqrt{6}}{4}\right)^2+\left(-\frac{5\sqrt{2}}{4}+\frac{\sqrt{82}}{4}\right)^2+2\left(-\frac{5}{4}+\frac{\sqrt{41}}{4}\right)^2+1}} & \dfrac{\frac{\sqrt{246}}{4}+\frac{7\sqrt{6}}{4}}{\sqrt{1+\left(\frac{5\sqrt{2}}{4}+\frac{\sqrt{82}}{4}\right)^2+2\left(\frac{5}{4}+\frac{\sqrt{41}}{4}\right)^2+\left(\frac{\sqrt{246}}{4}+\frac{7\sqrt{6}}{4}\right)^2}} & 0 & 0 & 0 \\[4mm]
0 & 0 & \dfrac{\sqrt{2}}{6} & 0 & -\dfrac{\sqrt{2}}{2} \\[2mm]
0 & 0 & \dfrac{\sqrt{2}}{6} & 0 & \dfrac{\sqrt{2}}{2} \\[2mm]
\dfrac{-\frac{5\sqrt{2}}{4}+\frac{\sqrt{82}}{4}}{\sqrt{\left(-\frac{\sqrt{246}}{4}+\frac{7\sqrt{6}}{4}\right)^2+\left(-\frac{5\sqrt{2}}{4}+\frac{\sqrt{82}}{4}\right)^2+2\left(-\frac{5}{4}+\frac{\sqrt{41}}{4}\right)^2+1}} & \dfrac{-\frac{\sqrt{82}}{4}-\frac{5\sqrt{2}}{4}}{\sqrt{1+\left(\frac{5\sqrt{2}}{4}+\frac{\sqrt{82}}{4}\right)^2+2\left(\frac{5}{4}+\frac{\sqrt{41}}{4}\right)^2+\left(\frac{\sqrt{246}}{4}+\frac{7\sqrt{6}}{4}\right)^2}} & 0 & 0 & 0 \\[4mm]
\dfrac{1}{\sqrt{\left(-\frac{\sqrt{246}}{4}+\frac{7\sqrt{6}}{4}\right)^2+\left(-\frac{5\sqrt{2}}{4}+\frac{\sqrt{82}}{4}\right)^2+2\left(-\frac{5}{4}+\frac{\sqrt{41}}{4}\right)^2+1}} & \dfrac{1}{\sqrt{1+\left(\frac{5\sqrt{2}}{4}+\frac{\sqrt{82}}{4}\right)^2+2\left(\frac{5}{4}+\frac{\sqrt{41}}{4}\right)^2+\left(\frac{\sqrt{246}}{4}+\frac{7\sqrt{6}}{4}\right)^2}} & 0 & 0 & 0 \\[4mm]
0 & 0 & \dfrac{2\sqrt{2}}{3} & 0 & 0 \end{bmatrix}$$

```
U, S, V = svd(convert(Matrix{Float64}, Φ));
```

U

```
5×5 Array{Float64,2}:
 0.605913  0.0       -0.364513  -0.707107      0.0
 0.605913  0.0       -0.364513   0.707107      0.0
 0.515499  0.0        0.85689    2.05391e-15   0.0
 0.0       0.707107   0.0        0.0          -0.707107
 0.0       0.707107   0.0        0.0           0.707107
```

S

```
5-element Array{Float64,1}:
 3.1147330734296355
 3.0
 1.8161602025381944
 1.4142135623730954
 1.0000000000000002
```

V

```
8×5 Adjoint{Float64,Array{Float64,2}}:
 0.275109  -0.0        -0.28384    -0.707107     -0.0
 0.275109  -0.0        -0.28384     0.707107     -0.0
 0.28666   -0.0         0.817206    2.83107e-15  -0.0
 0.0        0.235702    0.0        -0.0          -0.707107
 0.0        0.235702    0.0        -0.0           0.707107
 0.389062   0.0        -0.401411   -1.33227e-15   0.0
 0.784275   0.0         0.0995657   3.88578e-16   0.0
 0.0        0.942809    0.0         0.0           1.11022e-16
```

```
5×5 Array{Float64,2}:
 0.605913  0.0        -0.364513  -0.707107      0.0
 0.605913  0.0        -0.364513   0.707107      0.0
 0.515499  0.0         0.85689    2.05391e-15   0.0
 0.0       0.707107    0.0        0.0          -0.707107
 0.0       0.707107    0.0        0.0           0.707107
```
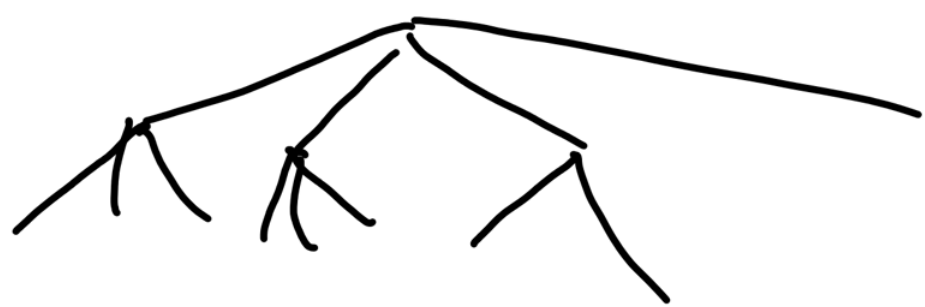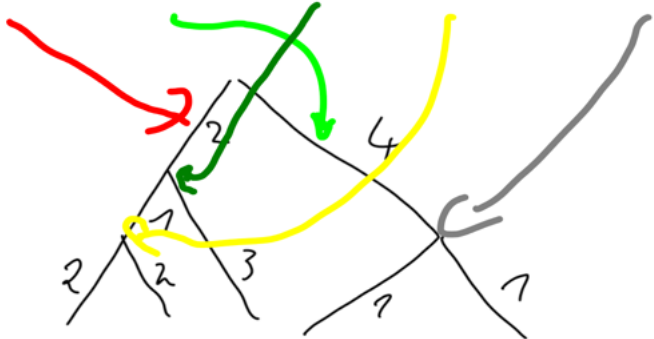


```
Φ = sy([
    1 0 0  0 0 0  0 0  0  1 0 0
    0 1 0  0 0 0  0 0  0  1 0 0
    0 0 1  0 0 0  0 0  0  1 0 0
    0 0 0  1 0 0  0 0  0  0 1 0
    0 0 0  0 1 0  0 0  0  0 1 0
    0 0 0  0 0 1  0 0  0  0 1 0
    0 0 0  0 0 0  1 0  0  0 0 1
    0 0 0  0 0 0  0 1  0  0 0 1
    0 0 0  0 0 0  0 0  1  0 0 0
])
```

$$\begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0
\end{bmatrix}$$

```
Φ * Φ.T
```

$$\begin{bmatrix} 2 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 2 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 2 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 2 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

```
U, S, V = sySVD(Φ);
```

```
U
```

$$\begin{bmatrix} 0 & \frac{\sqrt{3}}{3} & 0 & 0 & 0 & 0 & 0 & -\frac{\sqrt{2}}{2} & -\frac{\sqrt{2}}{2} \\ 0 & \frac{\sqrt{3}}{3} & 0 & 0 & 0 & 0 & 0 & 0 & \frac{\sqrt{2}}{2} \\ 0 & \frac{\sqrt{3}}{3} & 0 & 0 & 0 & 0 & 0 & \frac{\sqrt{2}}{2} & 0 \\ \frac{\sqrt{3}}{3} & 0 & 0 & 0 & 0 & -\frac{\sqrt{2}}{2} & -\frac{\sqrt{2}}{2} & 0 & 0 \\ \frac{\sqrt{3}}{3} & 0 & 0 & 0 & 0 & 0 & \frac{\sqrt{2}}{2} & 0 & 0 \\ \frac{\sqrt{3}}{3} & 0 & 0 & 0 & 0 & \frac{\sqrt{2}}{2} & 0 & 0 & 0 \\ 0 & 0 & \frac{\sqrt{2}}{2} & 0 & -\frac{\sqrt{2}}{2} & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{\sqrt{2}}{2} & 0 & \frac{\sqrt{2}}{2} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

```
S
```

$$\begin{bmatrix} 2 \\ 2 \\ \sqrt{3} \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$$
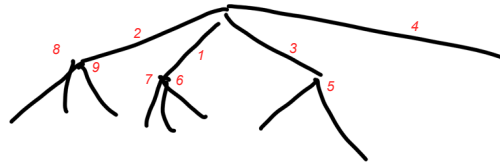
```
V
```

$$\begin{bmatrix}
0 & \frac{\sqrt{3}}{6} & 0 & 0 & 0 & 0 & 0 & -\frac{\sqrt{2}}{2} & -\frac{\sqrt{2}}{2} \\
0 & \frac{\sqrt{3}}{6} & 0 & 0 & 0 & 0 & 0 & 0 & \frac{\sqrt{2}}{2} \\
0 & \frac{\sqrt{3}}{6} & 0 & 0 & 0 & 0 & 0 & \frac{\sqrt{2}}{2} & 0 \\
\frac{\sqrt{3}}{6} & 0 & 0 & 0 & 0 & -\frac{\sqrt{2}}{2} & -\frac{\sqrt{2}}{2} & 0 & 0 \\
\frac{\sqrt{3}}{6} & 0 & 0 & 0 & 0 & 0 & \frac{\sqrt{2}}{2} & 0 & 0 \\
\frac{\sqrt{3}}{6} & 0 & 0 & 0 & 0 & \frac{\sqrt{2}}{2} & 0 & 0 & 0 \\
0 & 0 & \frac{\sqrt{6}}{6} & 0 & -\frac{\sqrt{2}}{2} & 0 & 0 & 0 & 0 \\
0 & 0 & \frac{\sqrt{6}}{6} & 0 & \frac{\sqrt{2}}{2} & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & \frac{\sqrt{3}}{2} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\frac{\sqrt{3}}{2} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & \frac{\sqrt{6}}{3} & 0 & 0 & 0 & 0 & 0 & 0
\end{bmatrix}$$

$$\begin{array}{ccccccccc}
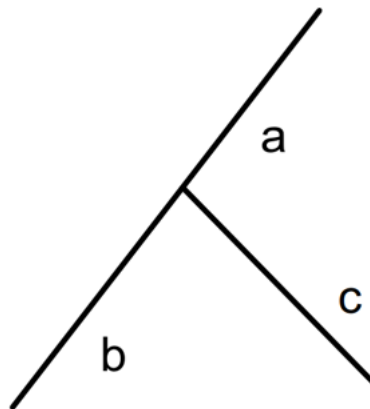1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9
\end{array}$$
$$\begin{bmatrix}
0 & \frac{\sqrt{3}}{3} & 0 & 0 & 0 & 0 & 0 & -\frac{\sqrt{2}}{2} & -\frac{\sqrt{2}}{2} \\
0 & \frac{\sqrt{3}}{3} & 0 & 0 & 0 & 0 & 0 & 0 & \frac{\sqrt{2}}{2} \\
0 & \frac{\sqrt{3}}{3} & 0 & 0 & 0 & 0 & 0 & \frac{\sqrt{2}}{2} & 0 \\
\frac{\sqrt{3}}{3} & 0 & 0 & 0 & 0 & -\frac{\sqrt{2}}{2} & -\frac{\sqrt{2}}{2} & 0 & 0 \\
\frac{\sqrt{3}}{3} & 0 & 0 & 0 & 0 & 0 & \frac{\sqrt{2}}{2} & 0 & 0 \\
\frac{\sqrt{3}}{3} & 0 & 0 & 0 & 0 & \frac{\sqrt{2}}{2} & 0 & 0 & 0 \\
0 & 0 & \frac{\sqrt{2}}{2} & 0 & -\frac{\sqrt{2}}{2} & 0 & 0 & 0 & 0 \\
0 & 0 & \frac{\sqrt{2}}{2} & 0 & \frac{\sqrt{2}}{2} & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0
\end{bmatrix}$$

Rule of thumb

There is one eigenvector/singular value for each daughter branch of the root. Additionally, for all non-root nodes with k daughters, there are $k-1$ eigenvectors/singular values.


```
@vars a b c
Φ = sy([
    sympy.sqrt(b)  0  sympy.sqrt(a)
    0  sympy.sqrt(c)  sympy.sqrt(a)
])
```

$$\begin{bmatrix} \sqrt{b} & 0 & \sqrt{a} \\ 0 & \sqrt{c} & \sqrt{a} \end{bmatrix}$$

```
U, D = (Φ*Φ.T).diagonalize()
U
```

$$\begin{bmatrix} -\dfrac{2a}{b-c+\sqrt{4a^2+b^2-2bc+c^2}} & \dfrac{2a}{-b+c+\sqrt{4a^2+b^2-2bc+c^2}} \\ 1 & 1 \end{bmatrix}$$

```
S = sqrt.(diag(D))
```

$$\begin{bmatrix} \sqrt{a + \frac{b}{2} + \frac{c}{2} - \frac{\sqrt{4a^2+b^2-2bc+c^2}}{2}} \\ \sqrt{a + \frac{b}{2} + \frac{c}{2} + \frac{\sqrt{4a^2+b^2-2bc+c^2}}{2}} \end{bmatrix}$$

Estimating the phylogenetic mean

The values x_t at the tips are observed. We want to estimate the value x_r at the root.

Recall that x_t is distributed according to a multivariate normal distribution with mean $x_r \mathbf{1}$ and covariance matrix $\Phi\Phi'$. According to the SVD of Φ, we have

$$\Phi = USV'$$
$$\Phi\Phi' = USV'VSU'$$
$$= (US)(US)'$$

Therefore the distribution of x_t can be represented as

$$z_i \sim \mathcal{N}(0,1)$$
$$x_t := USz + x_r\mathbf{1}$$

$\hat{x}_r$, the maximum likelihood estimate of x_r, is the value that maximizes the joint likelihood of z. This is equivalent of demanding that $z'z$ — the sum of squares of the components of z — is maximized.

$$USz = x_t - x_r\mathbf{1}$$
$$z = S^{-1}U'x_t - x_r S^{-1}U'\mathbf{1}$$
$$z'z = (S^{-1}U'x_t - x_r S^{-1}U'\mathbf{1})'(S^{-1}U'x_t - x_r S^{-1}U'\mathbf{1})$$
$$= (x_t'US^{-1} - x_r\mathbf{1}'US^{-1})(S^{-1}U'x_t - x_r S^{-1}U'\mathbf{1})$$
$$= x_t'US^{-2}U'x_t - x_r\mathbf{1}'US^{-2}U'x_t - x_r x_t'US^{-2}U'\mathbf{1} + x_r^2\mathbf{1}'US^{-2}U'\mathbf{1}$$
$$= x_t'US^{-2}U'x_t - 2x_r x_t'US^{-2}U'\mathbf{1} + x_r^2\mathbf{1}'US^{-2}U\mathbf{1}$$

Using calculus to minimize $z'z$, we have

$$\frac{dz'z}{dx_r} = \frac{x_t'US^{-2}U'x_t - 2x_r x_t'US^{-2}U'\mathbf{1} + x_r^2\mathbf{1}'US^{-2}U'\mathbf{1}}{dx_r}$$
$$= -2x_t'US^{-2}U'\mathbf{1} + 2x_r\mathbf{1}'US^{-2}U'\mathbf{1}$$

Setting the derivative to 0:

$$2\hat{x}_r\mathbf{1}'US^{-2}U'\mathbf{1} = 2x_t'US^{-2}U'\mathbf{1}$$
$$\hat{x}_r = \frac{x_t'US^{-2}U'\mathbf{1}}{\mathbf{1}'US^{-2}U'\mathbf{1}}$$

Estimating the rate of evolution

Suppose we know the phylogeny, but the unit of times of the branch lengths are not identical to the time unit of the Brownian motion. Then we have another parameter $\sigma \in \mathbb{R}^+$, the standard deviation of x after one unit of time.

Then the branch-wise Brownian motion steps are distributed as

$$b \sim \mathcal{N}(\mathbf{0}, \sigma^2\Lambda),$$

and x_t as

$$x_t \sim \mathcal{N}(x_r\mathbf{1}, \sigma^2\Phi\Phi')$$

The distribution of x_t can be represented as

$$z_i \sim \mathcal{N}(0, \sigma^2)$$
$$x_t := USz + x_r \mathbf{1}$$

Therefore

$$z = S^{-1}U'(x_t - x_r \mathbf{1})$$

The log-likelihood of z is

$$\mathcal{LL}(z|x_r, \sigma) = -\frac{n}{2}\log 2\pi - n\log \sigma - \frac{z'z}{2\sigma^2}$$

Then the log-likelihood of x_t is

$$\mathcal{LL}(x_t|x_r, \sigma) = -\frac{n}{2}\log 2\pi - n\log \sigma - \frac{z'z}{2\sigma^2} + \log\left|\frac{\partial z}{\partial x_t}\right|$$
$$= -\frac{n}{2}\log 2\pi - n\log \sigma - \frac{(x_t' - x_r \mathbf{1}')US^{-2}U'(x_t - x_r \mathbf{1})}{2\sigma^2} + \log|S|$$
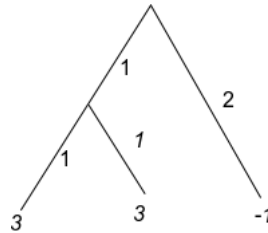
The gradient against x_r and σ is

$$\frac{\partial}{\partial x_r}\mathcal{LL}(x_t|x_r, \sigma) = \sigma^{-2}(x_t'US^{-2}U'\mathbf{1} - x_r\mathbf{1}'US^{-2}\mathbf{1})$$
$$\frac{\partial}{\partial \sigma}\mathcal{LL}(x_t|x_r, \sigma) = -\frac{n}{\sigma} + \frac{(x_t' - x_r\mathbf{1}')US^{-2}U'(x_t - x_r\mathbf{1})}{\sigma^3}$$

Setting the gradient to 0, we get the maximum likelihood estimates

$$\hat{x}_r = \frac{x_t'US^{-2}U'\mathbf{1}}{\mathbf{1}'US^{-2}U'\mathbf{1}}$$
$$\hat{\sigma} = \sqrt{\frac{(x_t' - \hat{x}_r\mathbf{1}')US^{-2}U'(x_t - \hat{x}_r\mathbf{1})}{n}}$$

Examples


```
Φ = sy([
    1 0 0 1
    0 1 0 1
    0 0 sympy.sqrt(2) 0
])
```

$$\begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & \sqrt{2} & 0 \end{bmatrix}$$

```
U, S, V = sySVD(Φ);
```

```
U
```

$$\begin{bmatrix} \frac{\sqrt{2}}{2} & 0 & -\frac{\sqrt{2}}{2} \\ \frac{\sqrt{2}}{2} & 0 & \frac{\sqrt{2}}{2} \\ 0 & 1 & 0 \end{bmatrix}$$

```
diagm(S)
```

$$\begin{bmatrix} \sqrt{3} & 0 & 0 \\ 0 & \sqrt{2} & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

```
xt = [3, 3, -1]
```

```
3-element Array{Int64,1}:
  3
  3
 -1
```

```
invCov = U * diagm(S.^-2) * U'
xr = (xt' * invCov * ones(Int, 3))/(ones(Int, 3)' * invCov*ones(Int, 3))
```

$\frac{9}{7}$

```
float(xr)
```

```
1.2857142857142858
```

```
residuals = xt .- xr
```

$$\begin{bmatrix} \frac{12}{7} \\ \frac{12}{7} \\ -\frac{16}{7} \end{bmatrix}$$

```
σ = sqrt((residuals' * invCov * residuals)/length(xt))
```

$\frac{4\sqrt{42}}{21}$

```
float(σ)
```

```
1.23442679969673528208875570211199936337241986801113628516448456725641405303631
```

```
float(residuals/σ)
```

```
3-element Array{Float64,1}:
  1.3887301496588271
  1.3887301496588271
 -1.8516401995451028
```

```
z = convert(Vector{Sym}, diagm(1 ./ S) * U' * (xt .- xr))
```

$$\begin{bmatrix} \frac{4\sqrt{6}}{7} \\ -\frac{8\sqrt{2}}{7} \\ 0 \end{bmatrix}$$

```
float.(z)
```

```
3-element Array{AbstractFloat,1}:
   1.3997084244475303418270194712605093668376842746609543591043957527148
34501404173
  -1.6162440712835371986305013990967978040796250004307977979162054148465
51403956686
   0.0
```
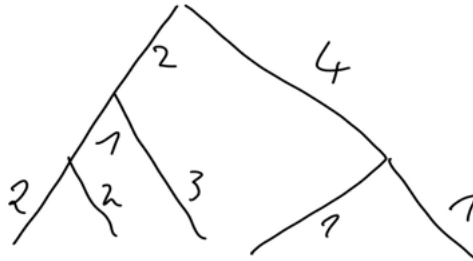
The phylogenetic mean is a weighted mean of the values at the tips. The weights only depend on Φ; they are given by the formula

$$w = \frac{US^{-2}U'\mathbf{1}}{\mathbf{1}'US^{-2}U'\mathbf{1}}$$

```
w = (invCov * ones(Int, 3)) / (ones(Int, 3)' * invCov * ones(Int, 3))
```

$$\begin{bmatrix} \frac{2}{7} \\ \frac{2}{7} \\ \frac{3}{7} \end{bmatrix}$$

Another example:


```
Φ = [
      sqrt(2) 0 0 0 0 1 sqrt(2) 0
      0 sqrt(2) 0 0 0 1 sqrt(2) 0
      0 0 sqrt(3) 0 0 0 sqrt(2) 0
      0 0 0 1 0 0 0 2
      0 0 0 0 1 0 0 2
      ]
```

```
5×8 Array{Float64,2}:
 1.41421  0.0      0.0      0.0  0.0  1.0  1.41421  0.0
 0.0      1.41421  0.0      0.0  0.0  1.0  1.41421  0.0
 0.0      0.0      1.73205  0.0  0.0  0.0  1.41421  0.0
 0.0      0.0      0.0      1.0  0.0  0.0  0.0      2.0
 0.0      0.0      0.0      0.0  1.0  0.0  0.0      2.0
```

```
invCov = inv(Φ*Φ')
w = (invCov * ones(Int, 5)) / (ones(Int, 5)' * invCov * ones(Int, 5))
```

```
5-element Array{Float64,1}:
 0.1753246753246753
 0.1753246753246753
 0.23376623376623382
 0.20779220779220783
 0.20779220779220783
```

Phylogenetic PCA

We assume that we are dealing with several characters that evolve according to a correlated Brownian motion along the branches of a phylogeny. This means that the evolution on each branch is drawn from a centered but correlated normal distribution.

Let Z be a $n \times k$ matrix, where n is the number of branches of the phylogeny and k the number of features. We assume that the rows of Z are stochastically independent, and each row is drawn from a multivariate normal distribution:

$$Z_i \sim \mathcal{N}(\mathbf{0}, \Sigma)$$

Φ is an $m \times n$ phylogenetic matrix, with the SVD

$$\Phi = USV'$$

V is the matrix of eigenvectors of $\Phi'\Phi$, so it is a permutation matrix.

$$(V'Z)_{ij} = \sum_a V'_{ia} Z_{aj}$$
$$= \sum_a V_{ai} Z_{aj}$$
$$E(V'Z)_{ij} = 0$$

So the rows of $V'Z$ are mutually independent and drawn from the same distribution as Z.

$$\mathrm{Cov}((V'Z)_{ij}, (V'Z)_{ik}) = E((V'Z)_{ij}, (V'Z)_{ik}) - E((V'Z)_{ij})E((V'Z)_{ik})$$
$$= E((V'Z)_{ij}, (V'Z)_{ik})$$
$$= E(\sum_a \sum_b (V_{ai} Z_{aj} V_{bi} Z_{bk}))$$
$$= \sum_a (V_{ai}^2 E(Z_{aj} X_{ak}))$$
$$= \sum_a (V_{ai}^2 \sigma_{jk})$$
$$= \sigma_{jk}$$

The observed data X form an $m \times k$-matrix (m being the number of tips and k the number of features). It is generated by the process

$$X_t := \mathrm{diag}(\vec{\sigma})\Phi Z + \vec{x}_r$$

According to the SVD theorem,

$$\mathrm{diag}(\vec{\sigma})^{-1}(X_t - \vec{x}_r) = \Phi Z$$
$$= USV'Z$$
$$S^{-1}U'\mathrm{diag}(\vec{\sigma})^{-1}(X_t - \vec{x}_r) = V'Z$$

We can estimate the phylogenetic means $\vec{x}_r$ and the rates of evolution $\vec{\sigma}$. After normalizing each of the columns of X_t (subtracting the phylogenetic mean and dividing by the rate of evolution), we apply the transformation $S^{-1}U'$ to obtain estimates for $V'Z$. Since the rows of this matrix are drawn from the same distribution as Z, we can apply regular PCA to this matrix.