**Report from Breakout J, Transformations:**

Why do we transform?  
    to find patterns (maybe we just need Random Forests, ML for this)  
        spatially agnostic methods are easier to implement, but explaining why more difficult  
    to find scale on which correlation makes sense  
What do we do when we don’t have gridded data on rectangle? (For gridded data we have: wavelets, Fourier)  
    needlets (for gridded data on sphere)  
    methods exist for wavelet-like transforms for nearest neighbors (Sweldon?)  
        but these can’t be interpreted as being on a fixed scale  
    for arbitrary domains, can use transformations  
Transformations on graphs: how do we do them?  
    wavelets (Sharpnack)  
    Laplacians  
    e.g. Laplacian smoothing of data on a graph, use sparsity  
Transformations of parameters instead of data  
    we can impose restrictions (e.g. Effi mentioned we can constrain the direction of the flow of a river for directed graphs)  
    what kinds of transformations will help?  
        reparameterization of AR process coefficients can help immensely  
        GMRFs, look at partial autocorrelations, sparsity  
How do we model nonlinear dynamics?  
    you cannot model this fully with covariances (El Nino year dependencies versus la Nina year dependencies are completely different)  
        you cannot model this well with GMRF because of teleconnections when predicting far into the future  
    Can you use multiscale models and relate the different scales together?  
        linear functions with interactions on multiple scales can cause nonlinearity.  Nonstationarity may be necessary  
Can we combine different transforms?  
    independent component analysis?  
        for non-Gaussian processes, but currently has limited success  
        no unique definition of it, and it performs poorly in Gaussian case  
    random projections  
Empirical mode decomposition  
    similar to EOF in that it’s difficult to interpret, analyze  
    not explored well in statistical literature  
Dynamic mode decomposition  
    added frequencies and damping rates  
    how does this match with EOFs and dynamic mode decomposition  
We need transforms that are spatially and temporally aware  
    this is a problem with EOFs, for instance  
        upweights clusters rather than down weights them  
        how can you compare PCs for different datasets  
            Peter C. mentioned a solution?  
How do you do ANOVA when your transformation dines’t preserve variance?  
    What about when it induces correlation in observations?  
        FT does this, e.g. tomography data, stripe errors