Principal Components Analysis with application to remote sensing image analysis

D G Rossiter

Cornell University, Section of Soi & Cropl Sciences

April 12, 2020

Cornell University
College of Agriculture and Life Sciences

Topic: Factor Analysis

A generic term for methods that consider the **inter-relations** between a set of variables.

- Often the set of **predictors** which might be used in a multiple linear regression.
  - Multivariate observations on same objects (e.g., soil samples)
  - **Remote sensing**: a set of **co-registered** images of a scene
    - all bands of one image
    - bands of multiple (co-registered) sensors
    - one band or band product (e.g., NDVI) of a **time-series** of images
- This is an analysis of the **structure** of the **multivariate feature space** covered by a set of variables.

Cornell University
College of Agriculture and Life Sciences

Uses of factor analysis in remote sensing

1. Discover **relations between images**, and possible **groupings** of them

2. Discover **groupings of pixels** in a set of images (→ classification)

3. **Interpret** the resulting groupings in terms of processes

4. Diagnose **multi-collinearity**, since images are usually correlated

   - determine which images are most correlated
   - quantify **redundancy**, find the **most informative subset** of images

5. For **data reduction** for model inputs; two approaches:

   - Identify **representative** images for a **minimum data set**
   - Compute **synthetic images**

Topic: Principal Components Analysis (PCA)

- The simplest form of factor analysis; a **data reduction** technique.

- Gives insight into the relation between a set of **variables** *within a dataset*
  - This is completely **data driven**; different sets of observations from the same population will give different relations
  - So, a **data mining** approach

- Gives insight into the relation between a set of **pixels** in the **multivariate space** spanned by the set of images

What does PCA do?

1. The **vector space** made up of the original observations (e.g., stack of pixel values in a set of images) is **projected** onto another vector space;

2. The new space has the **same dimensionality** as the original[1], i.e., there are as many variables in the new space as in the old;

3. In this space the new **synthetic images**, also called **principal components** are **orthogonal** to each other, i.e. **completely uncorrelated**;

4. The synthetic images are arranged in **decreasing order of variance explained**; and the total variance is unchanged;

5. The contribution of each original variable to each synthetic image is given;

6. Each observation can be **re-projected** into the new (PC) space, by its value of the synthetic images

[1] unless the original was rank-deficient

Mathematics: the data matrix

PCA is a direct calculation from a **matrix** constructed from the multivariate dataset.

$\mathbf{X}$: centred data matrix (i.e., difference from mean), where:

- **rows** are **observations** (e.g., pixels, observation locations, sampled individuals)
- **columns** are **variables** (e.g., reflectance in a band, pixel values) measured at each observation
- may **scale** by dividing values by the variable's sample standard deviation
  - standardized vs. unstandardized, see below

This gives the location of each observation in **multivariate attribute space**.

Cornell University
College of Agriculture and Life Sciences

Mathematics: the covariance or correlation matrix

The data matrix $\mathbf{X}$ is used to build a matrix that shows the relation between data items:

- $\mathbf{C} = \mathbf{X}^T\mathbf{X}$: the **covariance** (unscaled) or **correlation** (scaled) matrix
- this is symmetric and positive (semi-)definite, so has all real roots

Cornell University
College of Agriculture and Life Sciences

Why can the correlation matrix be misleading?

- The individual pairwise correlations do *not* take into account the degree to which *both* of the variables may be correlated to others

- In the case where both are highly correlated to a several others this is *apparent* correlation, which may not reflect a real process

- Solution: compute **partial correlations**
  - the bivariate correlation between the two **residuals** from linear regression of each variable on all the others, less the one with which to pair
  - this accounts for the "lurking" effect of other variables, and shows what correlation remains that can not be otherwise accounted for
  - (see example below)

Cornell University
College of Agriculture and Life Sciences

Mathematics: Eigen decomposition

The key insight is that the **Eigen decomposition**[2] of $\mathbf{C}$ orders the synthetic variables into descending amounts of variance, and ensures they are **orthogonal** (Hotelling 1933).

- Decompose a square, symmetric positive-definite matrix, e.g., the correlation matrix $\mathbf{C}$ formed from a data matrix such that $\mathbf{AC} = \lambda\mathbf{C}$
- **Eigenvalues**: a diagonal matrix λ; off-diagonals 0, i.e., no covariances, so orthogonal; **Eigenvectors**: the transformation matrix $\mathbf{A}$
- The **eigenvectors** provide a **coördinate transformation** such that the matrix multiplied by the diagonal **eigenvalues** matrix is the same as multiplication by a matrix made up of the eigenvectors
- Eigenvectors span an orthogonal **vector space** onto which we can **project** the original data.

[2] (German *eigen* $\approx$ English "own, belonging to oneself")

Cornell University
College of Agriculture and Life Sciences

Computation

- $|\mathbf{C} - \lambda \mathbf{I}| = 0$: a determinant to find the **eigenvalues** of the correlation matrix
  - these are sometimes called the **characteristic values**
  - their relative magnitude is the proportion of the original covariance explained

- Then the axes of the new space, the **eigenvectors** γ_j (one per dimension) are the solutions to $(\mathbf{C} - \lambda_j I)\gamma_j = \mathbf{0}$

- Obtain **synthetic variables** by projection: $\mathbf{Y} = \mathbf{PC}$ where $\mathbf{P}$ is the row-wise matrix of eigenvectors (rotations).

Details

In practice the system is solved by the Singular Value Decomposition (SVD) of the data matrix.

This is equivalent but more stable than directly extracting the eigenvectors of the correlation matrix.

Accessible explanations with worked examples:

- Davis, J. C. (2002). *Statistics and data analysis in geology*. New York: John Wiley & Sons.

- Legendre, P., & Legendre, L. (1998). *Numerical ecology*. Oxford: Elsevier Science.

Standardized vs. unstandardized – 1

Standardized each variable (e.g., reflectance in a band) has its mean subtracted (so $\overline{x_{.j}} = 0$) and is divided by its sample standard deviation (so $\sigma(x_{.j}) = 1$);

- All variables (e.g., bands) are **equally important**, no matter their absolute values or spreads;
- Gives **equal weight** to all variables;
- This is usually what we want if variables are measured on different scales.
  - e.g., multivariate measurements of soil constituents
  - e.g., co-registered images from different sensors

Standardized vs. unstandardized – 2

Unstandardized use the **original variables**, in their original scales of measurement; generally the means are also subtracted to centre the variables

- Variables with **wider spreads** (often due to measurement scale) are **more important**, since they contribute more to the original variance
- This preserves the importance of variables with **more variance = more information**
- E.g., sensor with different radiometric resolutions (so wider range of numeric values); higher resolution will have more weight
- Bands with more variability will have more weight – maybe we want this.

Potential difference between un/standardized!

Example: variables with three orders of magnitude difference in standard deviation (in original measurement scale):

First, **unstandardized** PCs:

```
        k.a         k.b         p.a     caco3.a     caco3.b      sand.b
206.1333028 172.1004094  53.2891699  25.5945396  24.9265695  20.3849822
        p.b      clay.b      clay.a      sand.a      silt.b      silt.a
 19.1486608  15.1824385  15.0226513  14.7909094  13.1101435  11.3642023
      cec.b       cec.a        oc.a        oc.b        ph.b        ph.a
  1.3351728   0.9138645   0.7702299   0.7265857   0.4260936   0.4153947
     dens.b      dens.a
  0.2721703   0.2313593

Importance of components:
                            PC1      PC2      PC3      PC4      PC5      PC6
Standard deviation     250.7955 100.7590 47.69829 34.18739 27.84061 16.31755
Proportion of Variance   0.8092   0.1306  0.02927  0.01504  0.00997  0.00343
Cumulative Proportion    0.8092   0.9398  0.96909  0.98413  0.99410  0.99753
```

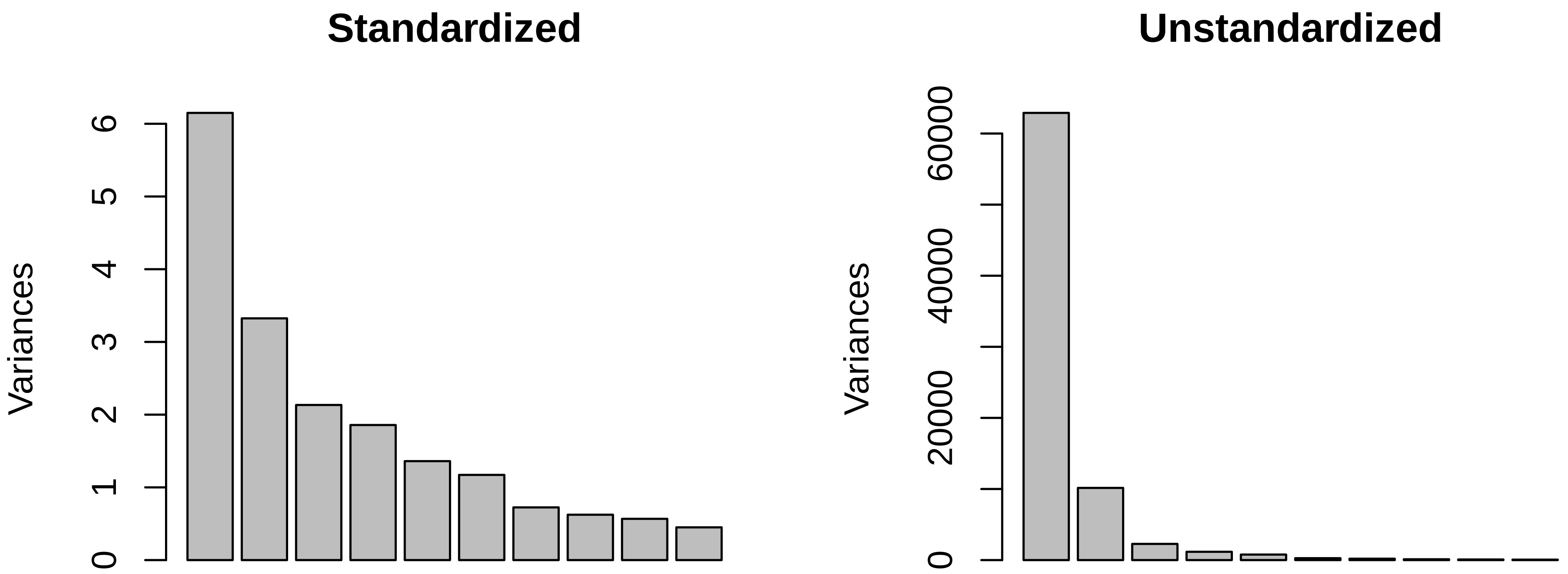
PC1 is numerically very large; K values have much larger standard deviations (higher absolute values of all measurements).

Second, **standardized** PCs

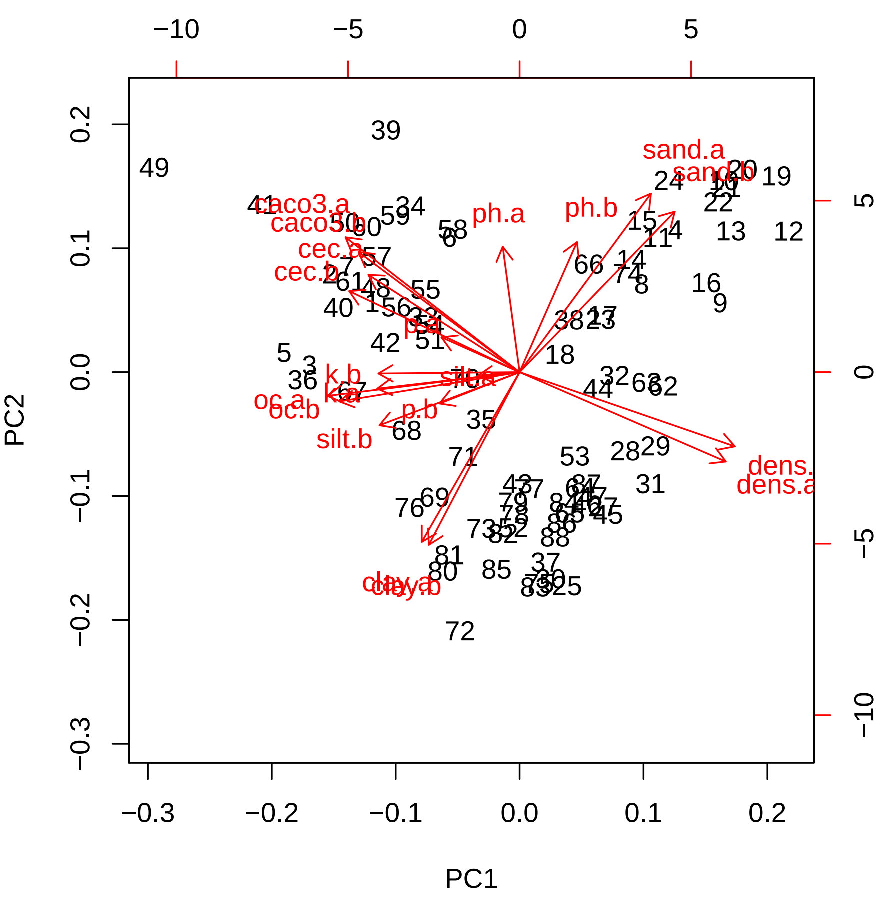
```
 dens.a  silt.a    ph.a   cec.a caco3.a     p.a  dens.b  sand.b  clay.b    oc.b
      1       1       1       1       1       1       1       1       1       1
  cec.b caco3.b     p.b  sand.a  clay.a    oc.a     k.a  silt.b    ph.b     k.b
      1       1       1       1       1       1       1       1       1       1

Importance of components:
                          PC1    PC2    PC3     PC4     PC5     PC6     PC7
Standard deviation     2.4798 1.8233 1.4605 1.36289 1.16650 1.08224 0.85127
Proportion of Variance 0.3075 0.1663 0.1067 0.09289 0.06805 0.05857 0.03624
Cumulative Proportion  0.3075 0.4738 0.5805 0.67336 0.74141 0.79998 0.83622
```

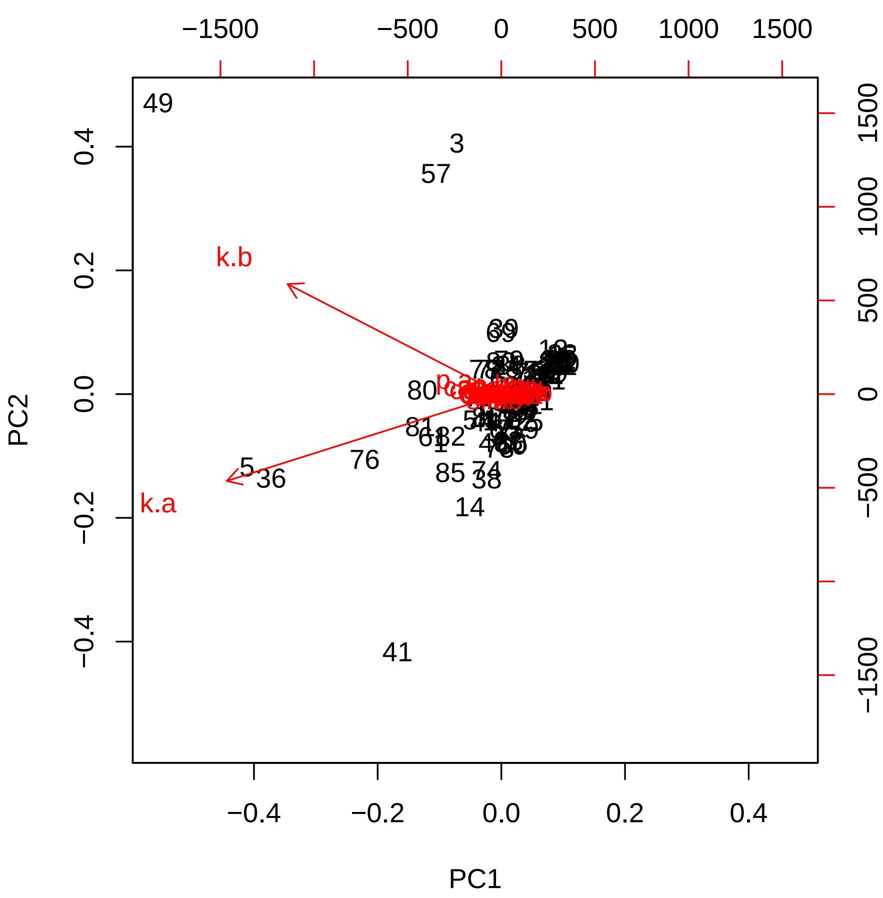
Same standard deviation (by design), much less total variance explained by PC1.

Difference in variance represented by PCs



Difference in biplots





Standardized
More or less equal loadings (lengths of arrows)

Unstandardized
K dominates loadings

(see below for explanation of biplots).

Topic: Remote sensing examples

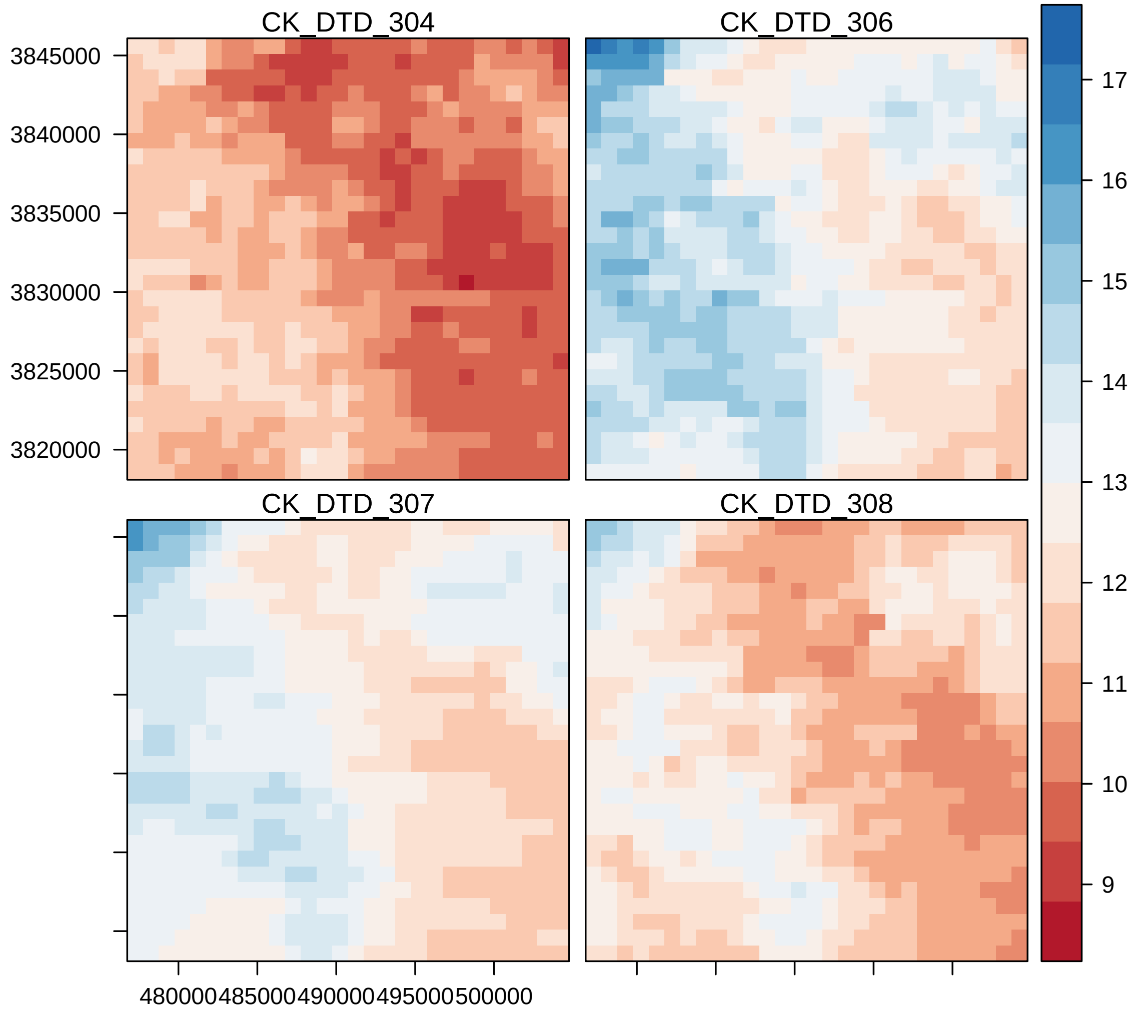
These will help visualize the transformation from original space into PC space.

Example: Time series of diurnal temperature differences

- 中国江苏沛县 Pei county in JiangSu province, PRC
- MODIS[3] daily land-surface temperature product[4]
- Images are **diurnal temperature differences** (day – night) between two MODIS products, units are Δ°C
- 30 Oct – 03 Nov 2000 (Julian days 304 ff.); Soil is drying after a heavy rain
- Objective: **try to relate DTD to soil texture and organic matter**
- Reference: Zhao, M.-S., Rossiter, D. G., Li, D.-C., Zhao, Y.-G., Liu, F., & Zhang, G.-L. (2014). *Mapping soil organic matter in low-relief areas based on land surface diurnal temperature difference and a vegetation index*. **Ecological Indicators**, 39, 120–133.[5]

[3] http://modis.gsfc.nasa.gov
[4] https://modis.gsfc.nasa.gov/data/dataprod/mod11.php
[5] http://doi.org/10.1016/j.ecolind.2013.12.015

Original images



Δ°C
(day - night)

Low DTD on first day after rain; increases overall then decreases (closer day/night air T?)

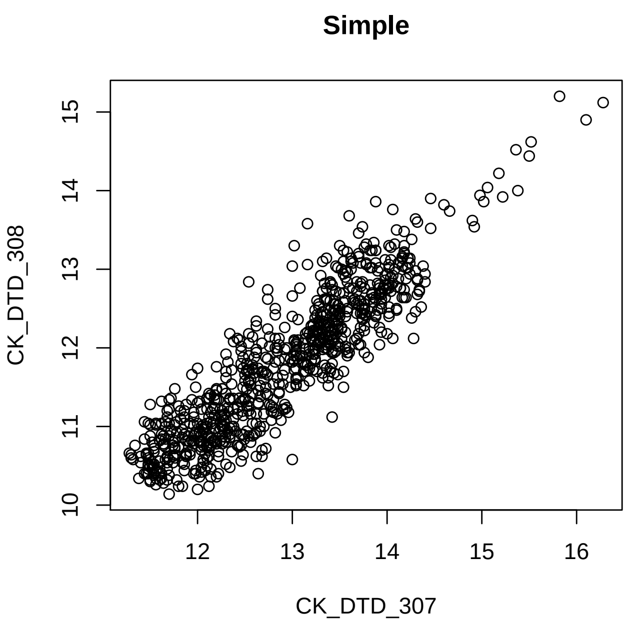
But: **similar overall pattern** of high vs. low DTD (**redundancy**)

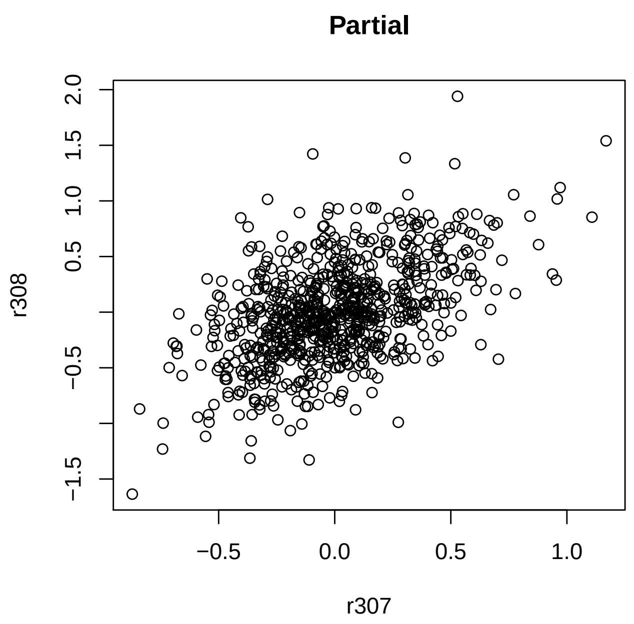
Reading in the dataset to R

```
## read images as a raster stack
require(raster)
(list <- dir(pattern='CK_DTD_30[0-9]{1}.img$'))
stackDTD <- stack(list, RAT=FALSE)
## add a Z value to represent the time
stackDTD <- setZ(stackDTD, c(304,306,307,308))
## PCA works with points
stackDTD.pts <- rasterToPoints(stackDTD)
stackDTD.df <- as.data.frame(stackDTD.pts)
```

Simple and partial correlations

```
> with(stackDTD.df, cor(CK_DTD_307, CK_DTD_308)) # simple correlation of two days
[1] 0.9029475
> # compute residuals from linear model on other days
> r307 <- residuals(lm(CK_DTD_307 ~ CK_DTD_304+CK_DTD_306, data=stackDTD.df))
> r308 <- residuals(lm(CK_DTD_308 ~ CK_DTD_304+CK_DTD_306, data=stackDTD.df))
> cor(r307, r308) # partial correlation = simple correlation of residuals
[1] 0.5054682
> plot(CK_DTD_308 ~ CK_DTD_307, data=stackDTD.df); plot(r308 ~ r307)
```





Partial correlation accounts for correlations of each with the other days

PCA processing in R

In this example we use **unstandardized** PCA because the four DTD images are on the **same scale** from the **same sensor** and represent the **same phenomenon**.

```
## PCA -- unstandardized, use original DTD, ignore coordinates
pca <- prcomp(stackDTD.pts[,3:dim(stackDTD.pts)[2]], scale = FALSE, retx=TRUE)
summary(pca) # show variance explained by each PC
pc$rotation # show contribution of each original band to each PC
screeplot(pca)

## extract synthetic bands, convert back to raster stack
stackDTD.scores <- cbind(stackDTD.pts[,1:2], pca$x)
stackDTD.scores <- data.frame(stackDTD.scores)
coordinates(stackDTD.scores) <- ~ x + y; gridded(stackDTD.scores) <- TRUE
stackDTD.scores <- stack(stackDTD.scores)
# name the synthetic bands in the raster stack
stackDTD.scores <- setZ(stackDTD.scores, paste("PC", 1:4, sep=""))
```


Cornell University
College of Agriculture and Life Sciences

Structure of prcomp object

```
> str(pca)
List of 5
 $ sdev    : num [1:4] 1.823 0.404 0.323 0.208
 $ rotation: num [1:4, 1:4] -0.459 -0.59 -0.466 -0.474 -0.675 ...
  ..- attr(*, "dimnames")=List of 2
  .. ..$ : chr [1:4] "CK_DTD_304" "CK_DTD_306" "CK_DTD_307" "CK_DTD_308"
  .. ..$ : chr [1:4] "PC1" "PC2" "PC3" "PC4"
 $ center  : Named num [1:4] 10.6 13.4 12.9 11.8
  ..- attr(*, "names")= chr [1:4] "CK_DTD_304" "CK_DTD_306" "CK_DTD_307" "CK_DTD_308"
 $ scale   : logi FALSE
 $ x       : num [1:784, 1:4] -6.17 -5.71 -4.64 -4.83 -4.43 ...
  ..- attr(*, "dimnames")=List of 2
  .. ..$ : NULL
  .. ..$ : chr [1:4] "PC1" "PC2" "PC3" "PC4"
 - attr(*, "class")= chr "prcomp"
```

rotation are the eigenvectors

x are the PC scores (location of each pixel in the PC space)

center are the image means (subtracted from all values)

PCA results – Importance of components

```
> summary(pca)
Importance of components:
                          PC1     PC2    PC3     PC4
Standard deviation     1.8232 0.40406 0.3230 0.20810
Proportion of Variance 0.9145 0.04492 0.0287 0.01191
Cumulative Proportion  0.9145 0.95938 0.9881 1.00000
```

The four DTD images are highly-correlated, 92% of the information is in common

I.e., over the four days the same areas tend to have narrow and wide DTD ranges

PCA results – rotations

```
> pc$rotation
           PC1     PC2     PC3     PC4
DTD304 -0.4447  0.5893  0.5870  0.3322
DTD306 -0.6084  0.3379 -0.5200 -0.4952
DTD307 -0.4717 -0.3857 -0.3677  0.7025
DTD308 -0.4578 -0.6243  0.4998 -0.3884
```

PC1 **“intensity”** of the phenomenon over all days – all signs the same (arbitrary), similar magnitudes

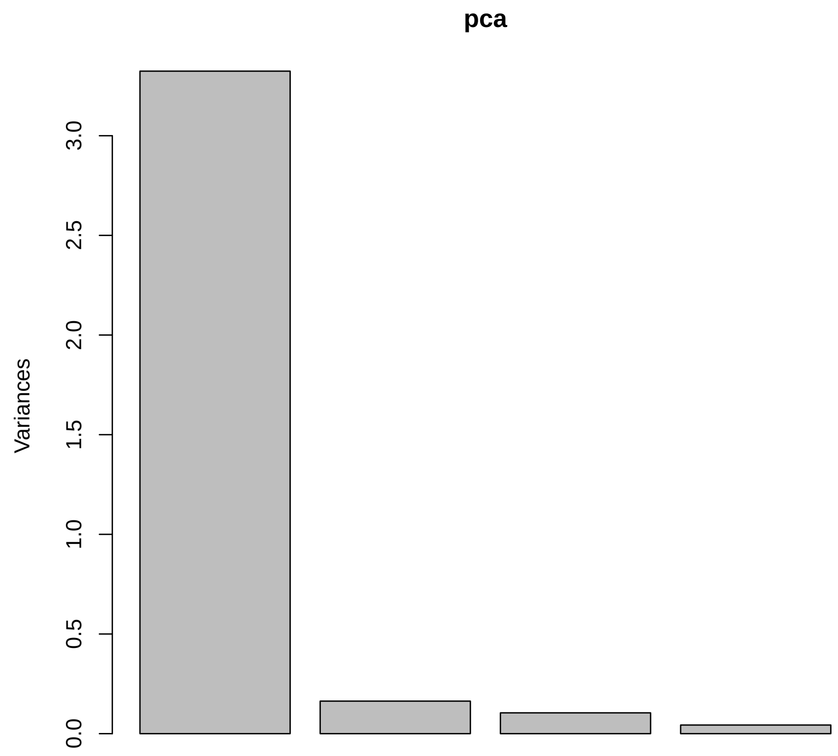
PC2 **contrast** between **first two and second two days**

PC3 **contrast** between **middle two and end two days**

PC4 Third and first days, contrasted with second and fourth days

Note PCs are **orthogonal** (independent)

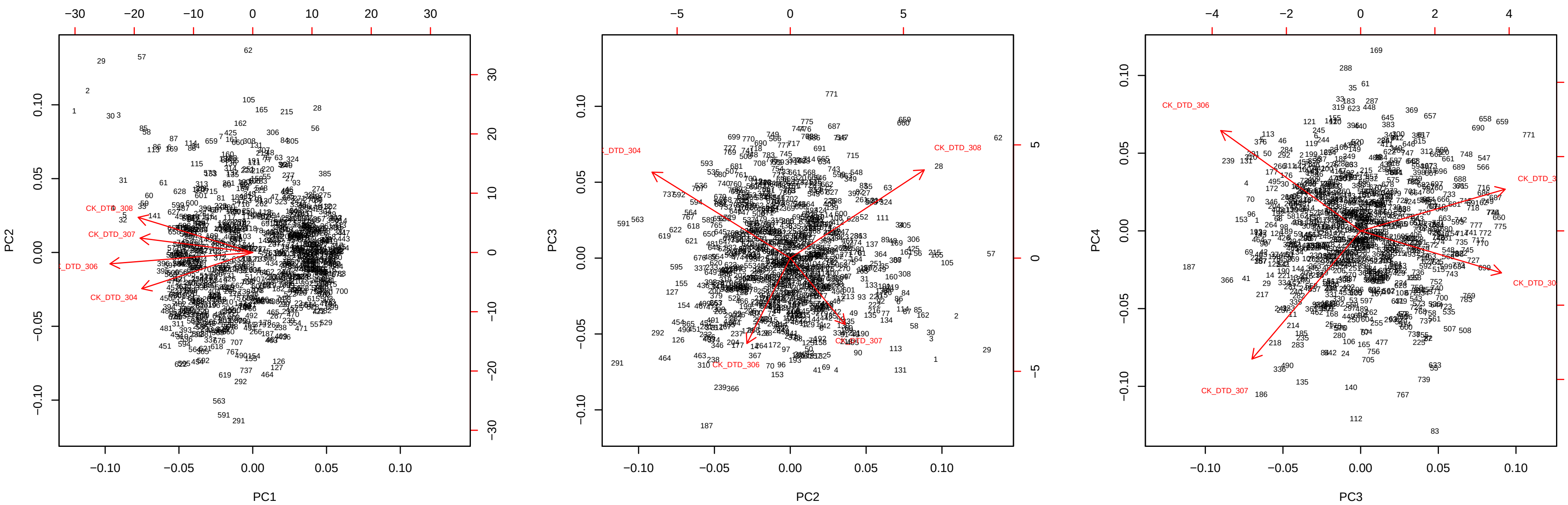
Screeplot



Biplots

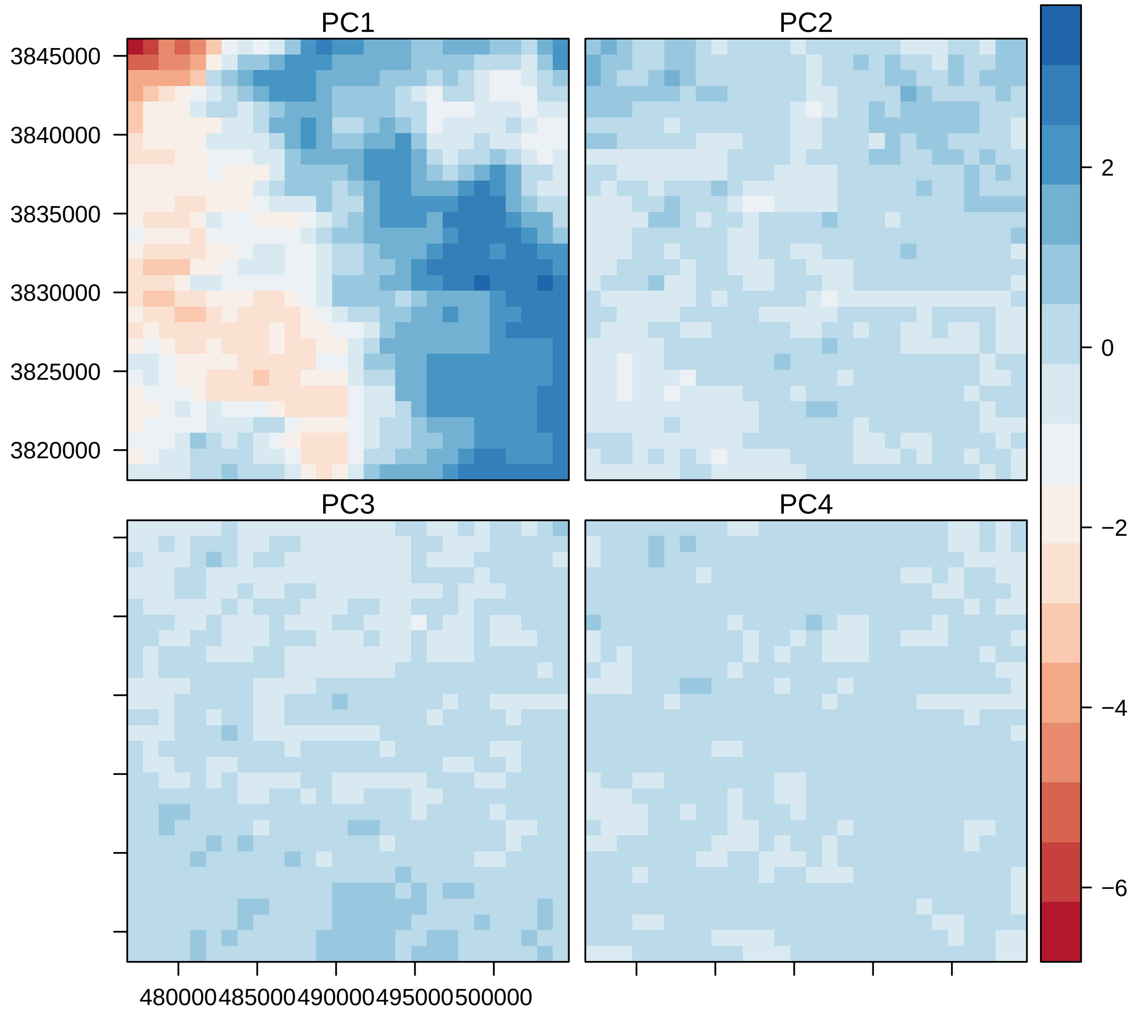
These show **scores** of the observations as synthetic variables in the 2-PC space (observation numbers)

Loadings of each variable in the 2-PC space shown by the arrows (longer = higher loading).



First PC is **overall intensity** across all 4 dates; other PCs show **contrasts** between dates

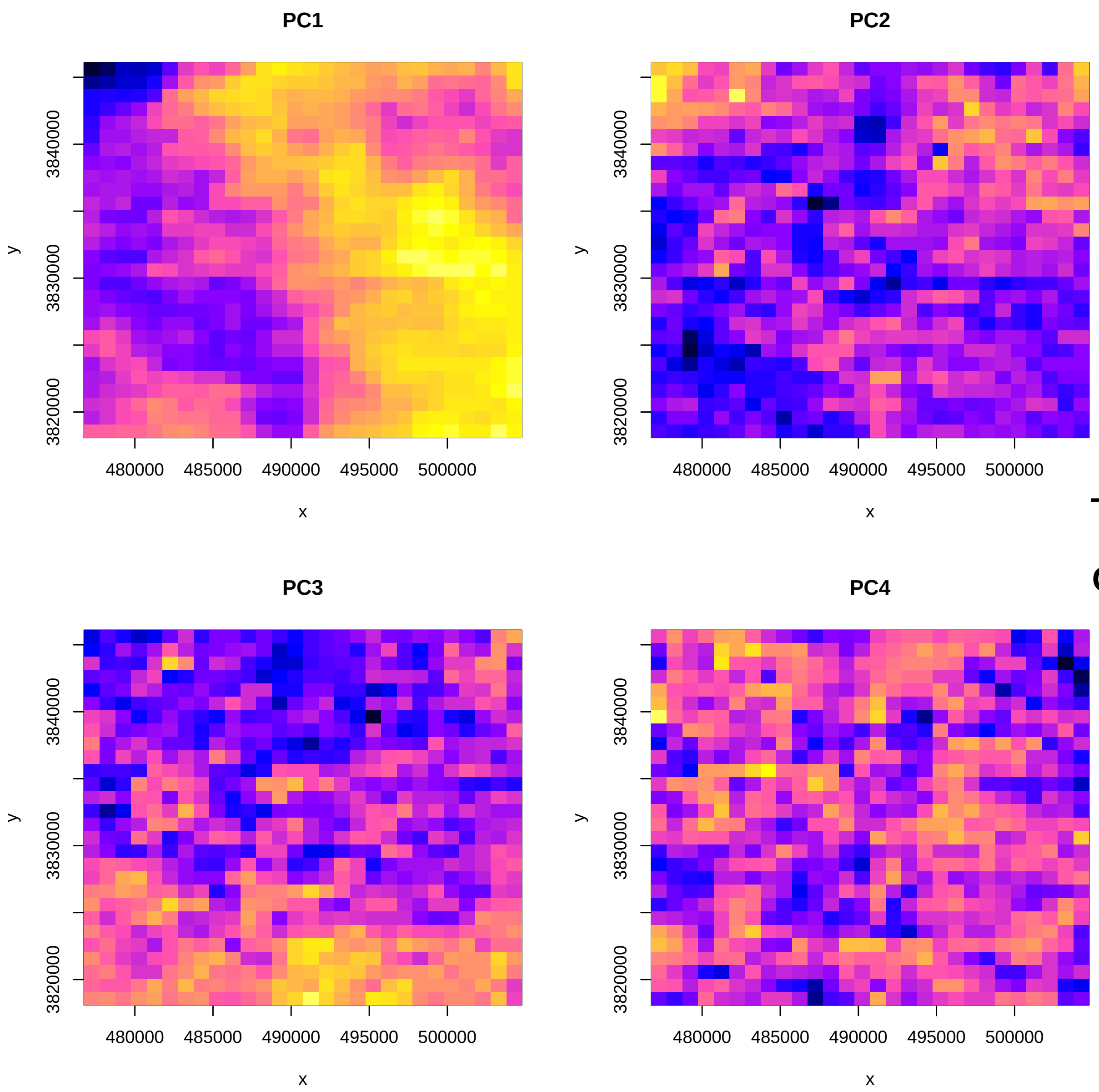
PCs (synthetic bands) – same stretch



Note **decreasing information content** as PC number increases

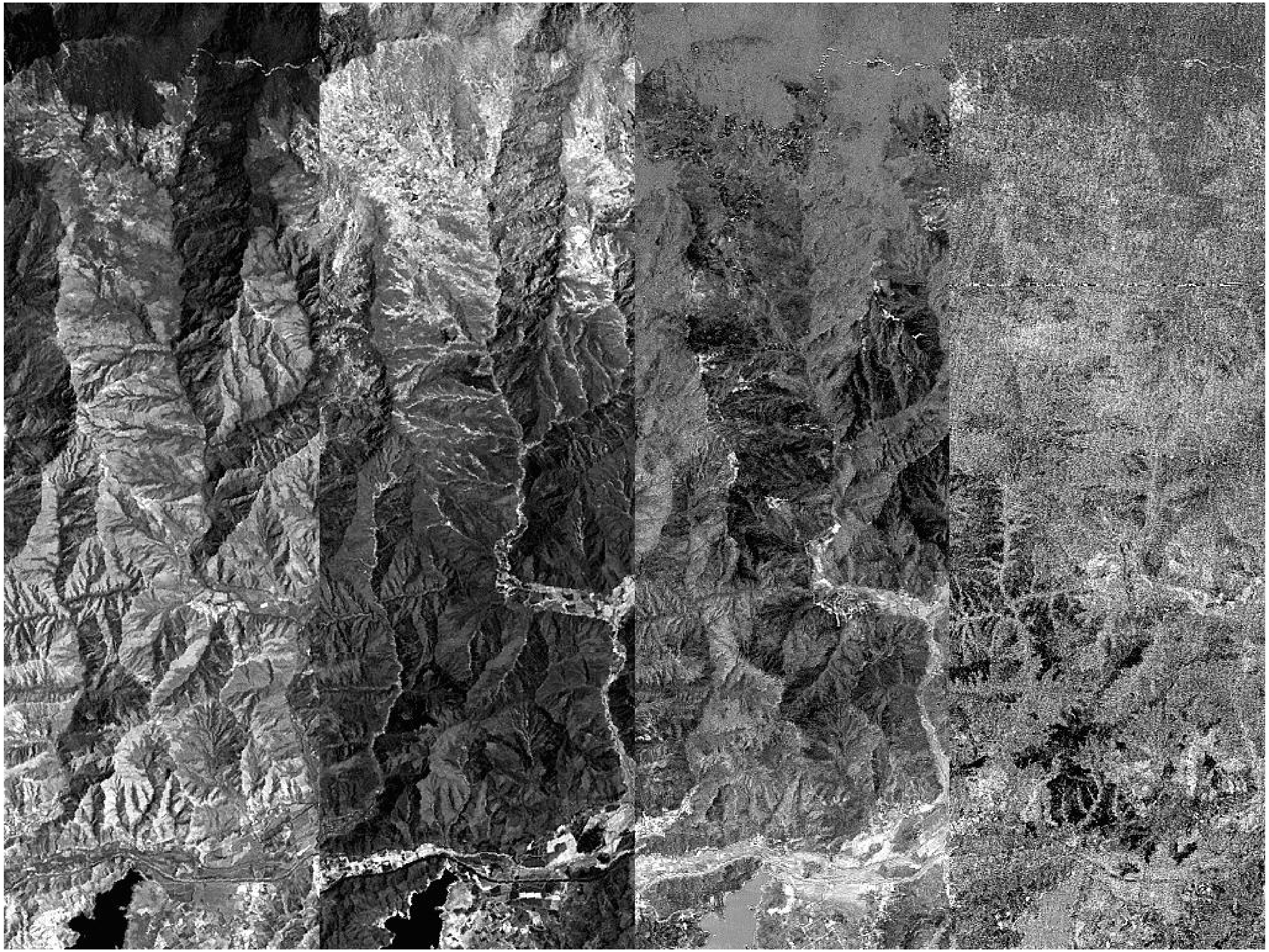
There is still some information in PC2, 3, 4 that **contrasts** with the overall pattern

PCs (synthetic bands) – individual stretch



This allows to visualize contrasts **within** a PC

Another example of synthetic images



Cordillera de la Costa, W branch Río Aragua, Aragua state, Venezuela

Conclusion

- PCA is a powerful data reduction technique
- It is linear – so if variables are highly skewed or multi-model, apply a transformation
- It can also reveal the inter-relation between variables (e.g., reflectances in various spectral bands)
- It is completely data-driven and is scene-specific
- An important choice is between standardized and unstandardized

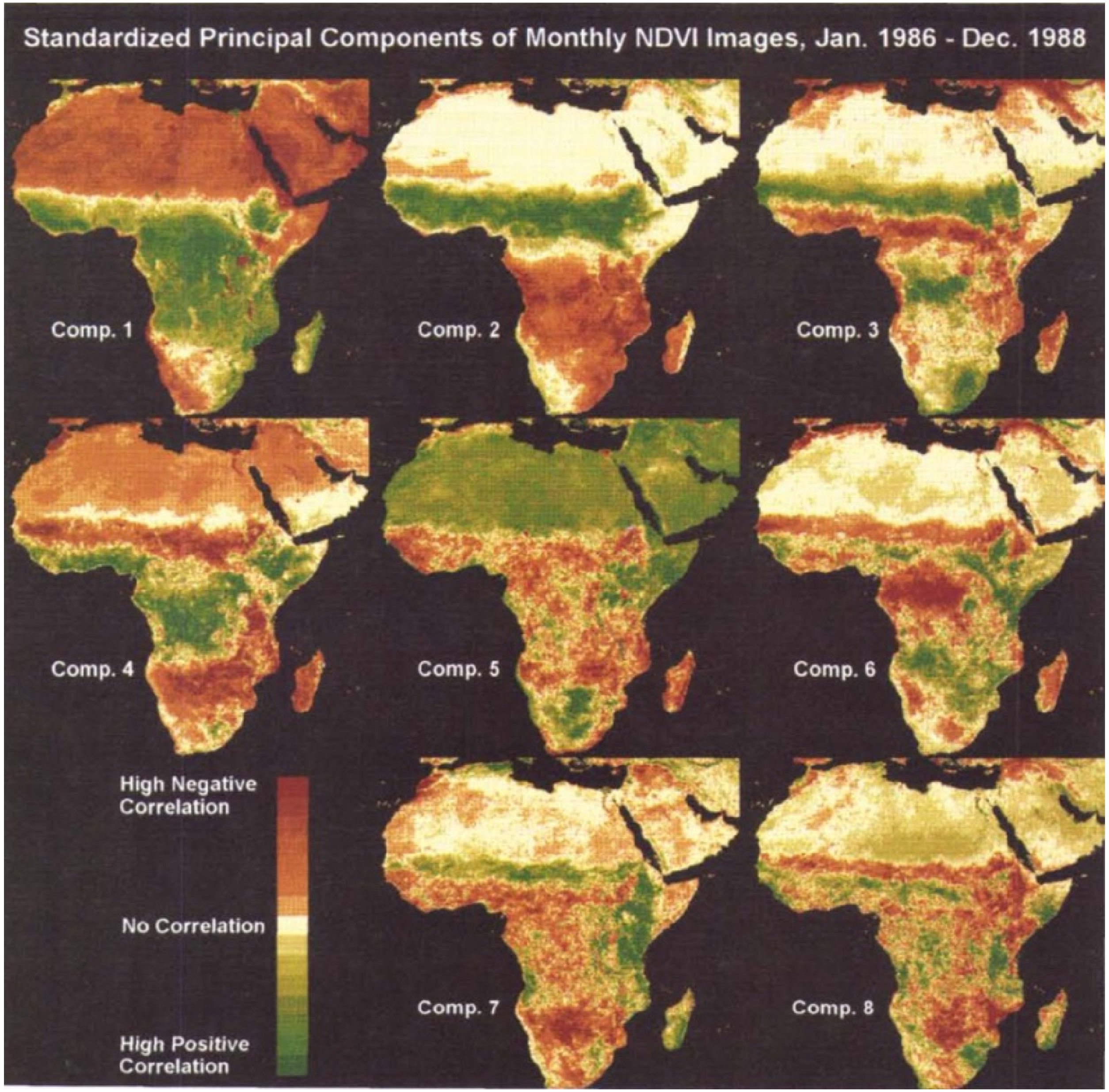
Topic: PCA of long time series of imagery to reveal seasonality

Groundbreaking paper, 200+ citations:

Eastman, J., & Fulk, M. (1993). *Long Sequence Time-Series Evaluation Using Standardized Principal Components* (reprinted from Photogrammetric Engineering and Remote-Sensing , Vol 59, Pg 991–996, 1993). **Photogrammetric Engineering and Remote Sensing**, 59(8), 1307–1312.

- Sensor was AVHRR
- Images were monthly maximum NDVI, 1986-1988 (three full years)

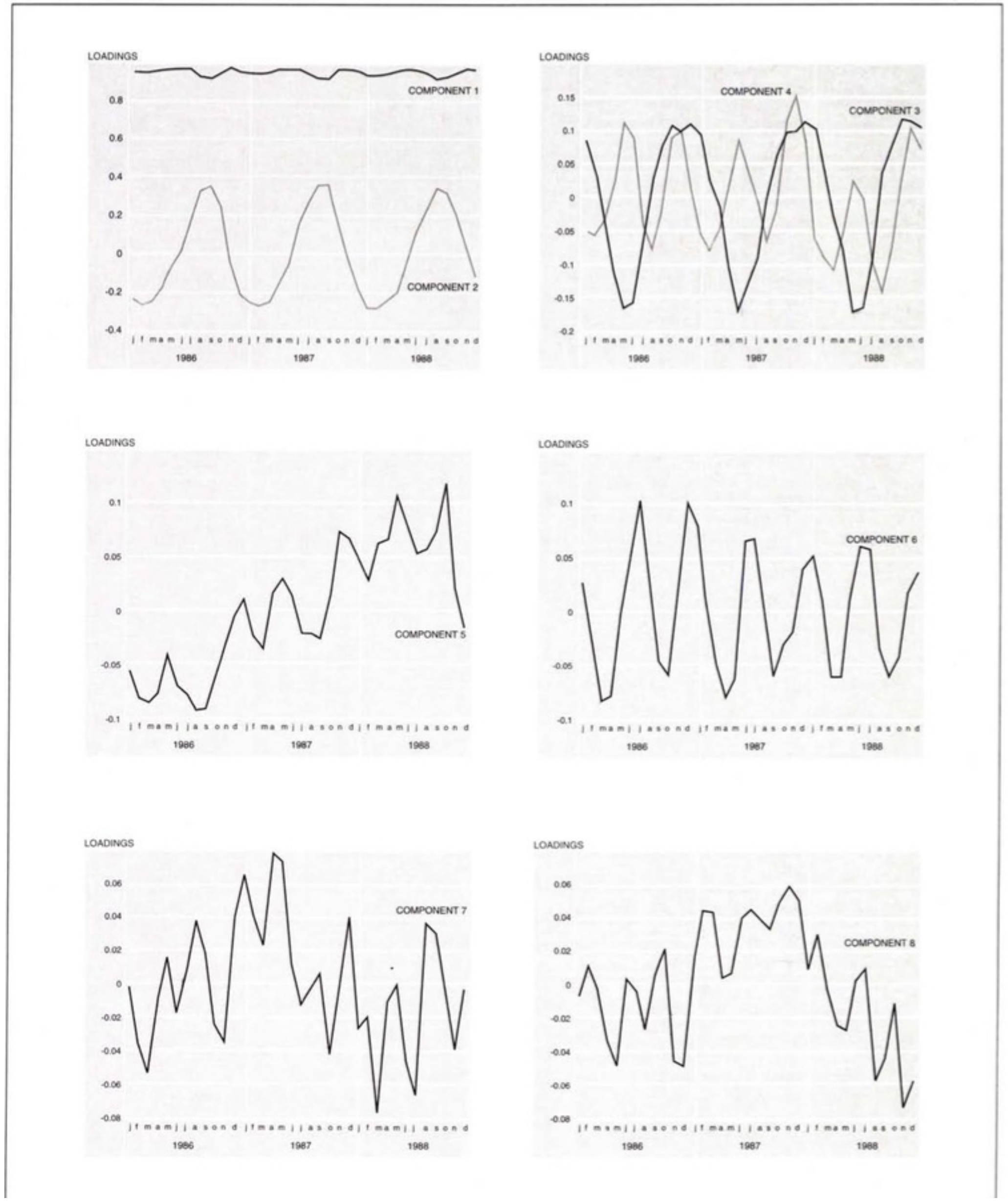
Synthetic images



Green = “+” score, Red = “-”

Sign is arbitrary, here chosen to show high NDVI in green.

Loadings



PC1 All bands contribute ≈ equally:(overall intensity)

PC2 shows annual movement of Intertropical Convergence Zone

PC3/4 show deviations from PC2 seasonality

Interpretation

Synthetic bands images summarizing all original images, according to the PCs

Loadings relation between original images (dates, x-axis) and contribution to the PC (y-axis).

- Note decreased absolute loadings at higher PCs, this is because they represent less of the total variability of the 36 images

- PC1: $\approx$ equal contribution of all dates (overall vegetation vigour averaged over the 3 years)

- PC2: + correlation in N. hemisphere summer, - in winter (seasonality) – Intertropical Convergence Zone

- PC3/4: deviations from PC2 seasonality – note time lag of greening/senescence

- PC5: detecting sensor drift

Topic: PCA for a multi- to hyper-variate dataset

A small dataset of 30 soil properties observed at 87 locations in the Lake Valencia basin, Venezuela

Also recorded coördinates (not used here) and geomorphic region

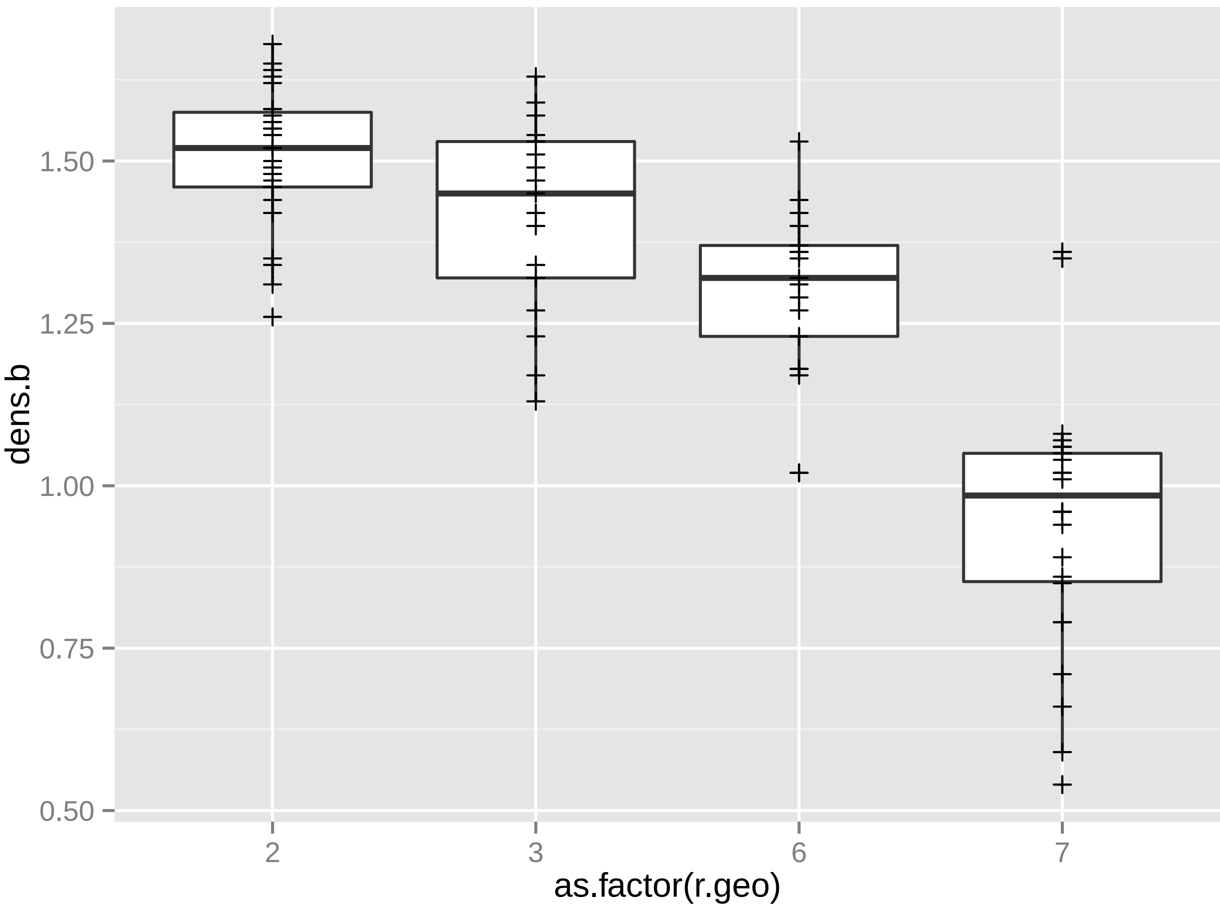
Main interest is to see the **inter-relation between variables** (grouping, contrasts)

- Could we only measure surface soil properties w.o. loss of informtion?
- Could we omit some (expensive?) measurements w.o. loss of informtion?

```
> dim(dlv.r)
[1] 87 33
> names(dlv.r)
 [1] "utm.n"   "utm.e"   "r.geo"   "dens.a"  "vcs.a"   "cs.a"    "ms.a"    "fs.a"
 [9] "vfs.a"   "sand.a"  "silt.a"  "clay.a"  "oc.a"    "ph.a"    "cec.a"   "caco3.a"
[17] "p.a"     "k.a"     "dens.b"  "vcs.b"   "cs.b"    "ms.b"    "fs.b"    "vfs.b"
[25] "sand.b"  "silt.b"  "clay.b"  "oc.b"    "ph.b"    "cec.b"   "caco3.b" "p.b"
[33] "k.b"
```

Geomorphic regions explain some variation

```
> ## boxplot of bulk density of the subsoil, by geomorphic region
> require(ggplot2)
> qplot(x=as.factor(r.geo), y=dens.b, data=dlv.r, geom=c("boxplot", "point"), shape=I(3))
```



Cornell University
College of Agriculture and Life Sciences

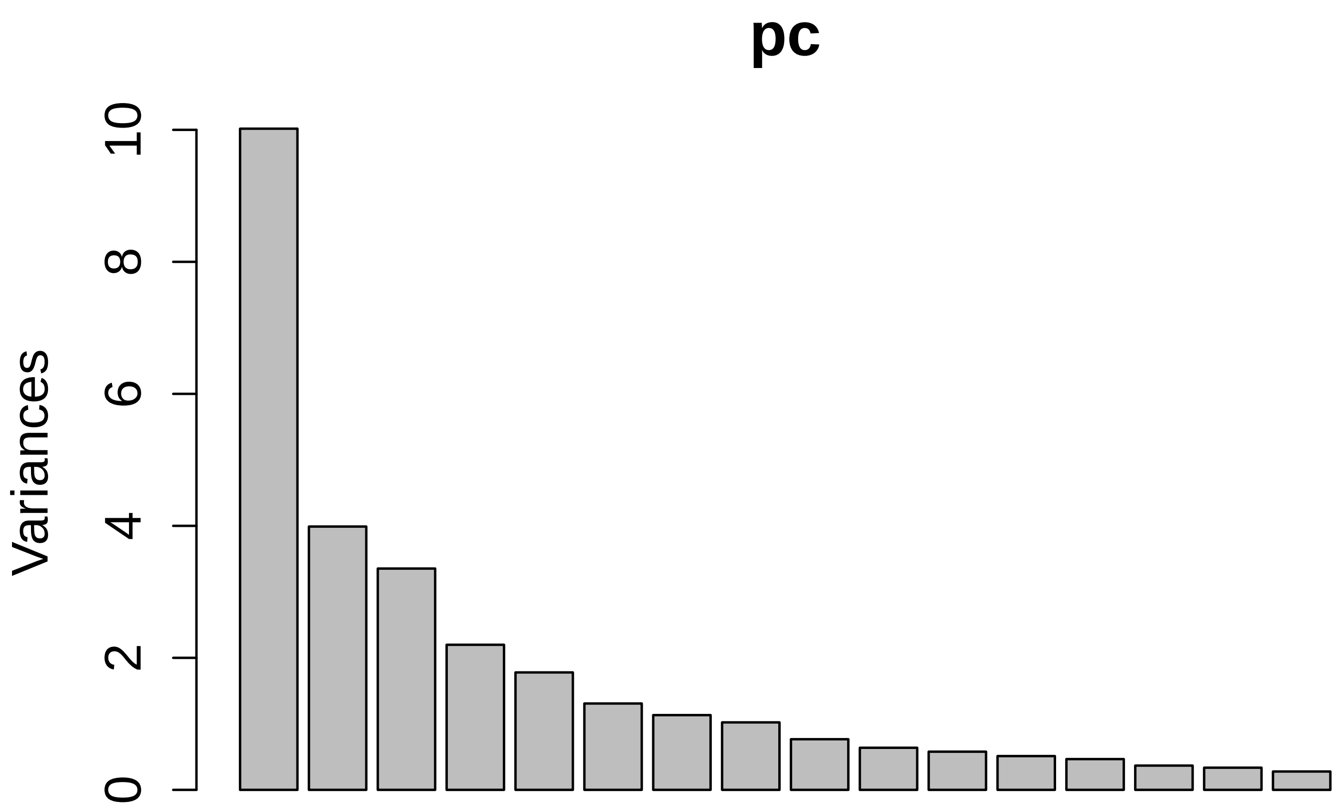
Computing standardized PCs

```
> pc <- prcomp(dlv.r[4:33], center=TRUE, scale.=TRUE)
> summary(pc)
Importance of components:
                          PC1    PC2    PC3     PC4     PC5     PC6     PC7     PC8
Standard deviation     3.1651 1.9973 1.8312 1.48272 1.33402 1.14400 1.06427 1.01107
Proportion of Variance 0.3339 0.1330 0.1118 0.07328 0.05932 0.04362 0.03776 0.03408
Cumulative Proportion  0.3339 0.4669 0.5787 0.65195 0.71127 0.75490 0.79265 0.82673
                           PC9    PC10    PC11    PC12    PC13    PC14   PC15    PC16
Standard deviation     0.87600 0.79861 0.76065 0.71565 0.68304 0.60664 0.5797 0.52760
Proportion of Variance 0.02558 0.02126 0.01929 0.01707 0.01555 0.01227 0.0112 0.00928
Cumulative Proportion  0.85231 0.87357 0.89285 0.90992 0.92548 0.93774 0.9489 0.95822
                          PC17    PC18    PC19    PC20    PC21   PC22    PC23    PC24
Standard deviation     0.50731 0.43862 0.42948 0.38868 0.33886 0.3146 0.28247 0.25723
Proportion of Variance 0.00858 0.00641 0.00615 0.00504 0.00383 0.0033 0.00266 0.00221
Cumulative Proportion  0.96680 0.97322 0.97936 0.98440 0.98823 0.9915 0.99418 0.99639
                          PC25   PC26    PC27    PC28    PC29    PC30
Standard deviation     0.20782 0.1647 0.14647 0.11458 0.04917 0.03113
Proportion of Variance 0.00144 0.0009 0.00072 0.00044 0.00008 0.00003
Cumulative Proportion  0.99783 0.9987 0.99945 0.99989 0.99997 1.00000
```

12 PCs (out of 30) explained **90%** of the standardized variance of **30** standardized variables

Screeplot

```
> screeplot(pc, npcs=16)
```



Synthetic variables

The standardized variables are multiplied by these factors to make up the synthetic variables (PCs)

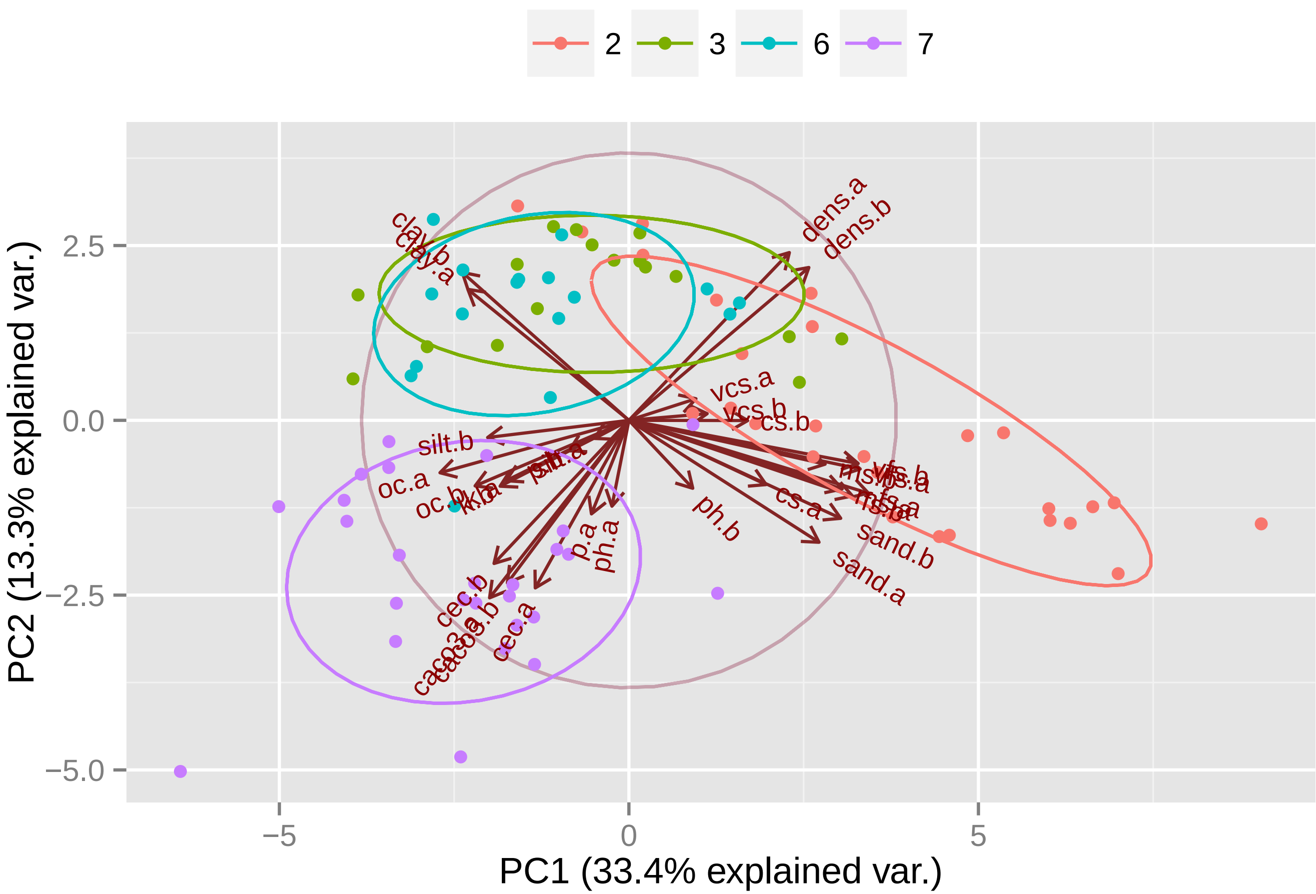
```
> print(pc$rotation[,1:4])
                PC1           PC2          PC3          PC4
dens.a   0.18958475  0.3139849696 -0.096232326  0.088356551
vcs.a    0.07924278  0.0403831655  0.230370094 -0.048419791
cs.a     0.16208849 -0.1202754319  0.363705662 -0.078683279
ms.a     0.25194385 -0.1284850503  0.210080111  0.008324169
fs.a     0.28293845 -0.1363107368 -0.014925221  0.054220372
vfs.a    0.27325614 -0.0910810904 -0.064745942  0.041049247
sand.a   0.22462929 -0.2282354263 -0.122083768  0.004537925
silt.a  -0.04055455 -0.0339314591  0.094532854 -0.128362638
clay.a  -0.19025200  0.2451180696  0.037298357  0.096288845
oc.a    -0.22319928 -0.0981217660  0.117350816 -0.103240123
ph.a    -0.02002915 -0.1607898185 -0.341449972 -0.032629362
cec.a   -0.11061555 -0.3136921769  0.084300743  0.073849593
caco3.a -0.16458980 -0.3322511108 -0.087971037 -0.214771086
p.a     -0.04422812 -0.1751203044  0.041317218  0.493486269
k.a     -0.14509102 -0.1129631442  0.068684059  0.389384538
dens.b   0.21263155  0.2860476925 -0.081790307  0.119293244
vcs.b    0.09246395  0.0124449952  0.416921421 -0.096156354
cs.b     0.14040571 -0.0003804787  0.390813767 -0.054448951
ms.b     0.23184462 -0.0820839309  0.213227766 -0.002106178
fs.b     0.26409667 -0.1415713026 -0.048437418  0.072954308
vfs.b    0.27090087 -0.0808317887 -0.116662538  0.078520254
sand.b   0.25024710 -0.1834095557 -0.092293422  0.045293178
silt.b  -0.16691593 -0.0321539033  0.149702956 -0.016192942
```

```
clay.b  -0.19685387  0.2769034074 -0.000293272 -0.046612720
oc.b    -0.18170694 -0.1227445391  0.191119295  0.086771096
ph.b     0.07542964 -0.1271823250 -0.326496078 -0.036204018
cec.b   -0.15933624 -0.2679633668 -0.019482653 -0.050358586
caco3.b -0.14353695 -0.3045767375 -0.026160011 -0.290825175
p.b     -0.07072839 -0.0569701860  0.100738984  0.438684722
k.b     -0.15217588 -0.1233052242 -0.006211632  0.404872772
```

Biplot

```
> plot.it <- function(which) {
+ g <- ggbiplot(pc, choice=which, obs.scale = 1, var.scale = 1,
              groups = r.geo.f, ellipse = TRUE,  circle = TRUE)
+ g <- g + scale_color_discrete(name = '')  + theme(legend.direction = 'horizontal',
+               legend.position = 'top')
+ print(g)}
> plot.it(1:2)
> plot.it(3:4)
```

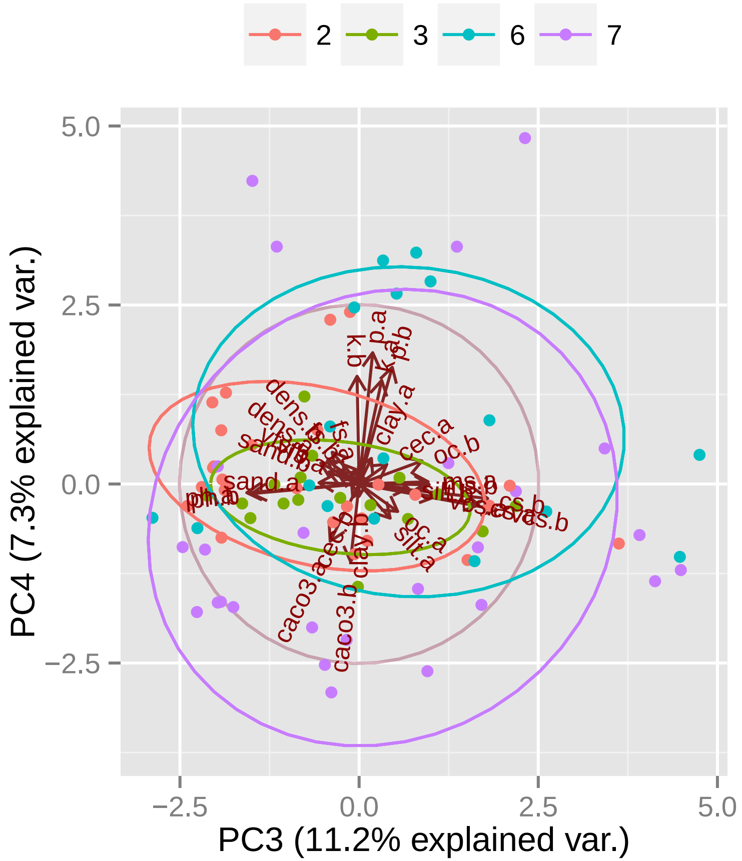
(see next slides)



- normal ellipse of each group in PC space; 1 s.d. by default
- distance between the points ≈ Mahalanobis distance; inner product between the variables ≈ correlation
- graph produced with `ggbiplot` package

Interpretation of PC1 and 2

- strong correlations between top/sub for most properties
- sand fractions highly correlated
- clay opposite sand
- bulk density opposite CEC/OC/silt/$CaCO3$
- PC1 +dense, sandy, low silt, low OC
- PC2 +low fertility, base saturation, carbonates
- But these are not exactly aligned with the PCs
- Geomorphic regions cluster points in PC1/2 space
- especially 2 (piedmont) and 7 (recently-emerged lake sediments)



- PC3 contrasts top and subsoil sand
- PC4 contrasts K, P with CaCO3

Topic: Factor analysis: beyond PCA

Limitations of PCA:

- PCA is a **data reduction** technique
- It does not impose any **structure** on the PCs or synthetic variables, they come out directly from the eigen decomposition of the correlation or covariance matrix
- So, the resulting PCs or synthetic variables may not be easily **interpretable**
  - the loadings may not line up with any of the axes – see Lake Valencia example
  - Clear associations of variables but none lined up with a PC

Latent variable analysis – concept

- “Factor analysis” as used in social sciences

- Hypothesis: the set of **observed** variables is a measureable expression of some (smaller) number of **latent** variables
  - These can not be directly measured,
  - They influence a number of the observed variables.
  - Example: “math ability”, “ability to think abstractly” are *assumed* to exist (based on external evidence or theory) and measured with various tests (observed variables).

- The set of observed variables can be analyzed with PCA and then (1) reduced; (2) **rotated** into interpretable components

Cornell University
College of Agriculture and Life Sciences

Latent variable analysis – computation

- from k **observed** variables hypothesize $p < k$ **latent** variables ("factors")

- decompose $k \times k$ variance-covariance matrix Σ of the original variables into:

  1. a $p \times k$ loadings matrix Λ (the k columns are the original variables, the p rows are the latent variables);
  2. and a $k \times k$ diagonal matrix of **unexplained variances** per original variable (its *uniqueness*) Ψ, so $\Sigma = \Lambda'\Lambda + \Psi$

- In PCA $p = k$, there is no Ψ, and all variance is explained by the synthetic variables; there is only one way to do this.

- In factor analysis, the loadings matrix Λ is *not unique*

  – it can be multiplied by any $k \times k$ orthogonal matrix, known as **rotations**.
  – The factor analysis algorithm finds a rotation to satisfy user-specified conditions; e.g., *varimax*.

Cornell University
College of Agriculture and Life Sciences

Latent variables – example

Lake Valencia, topsoils only; 15 observed variables

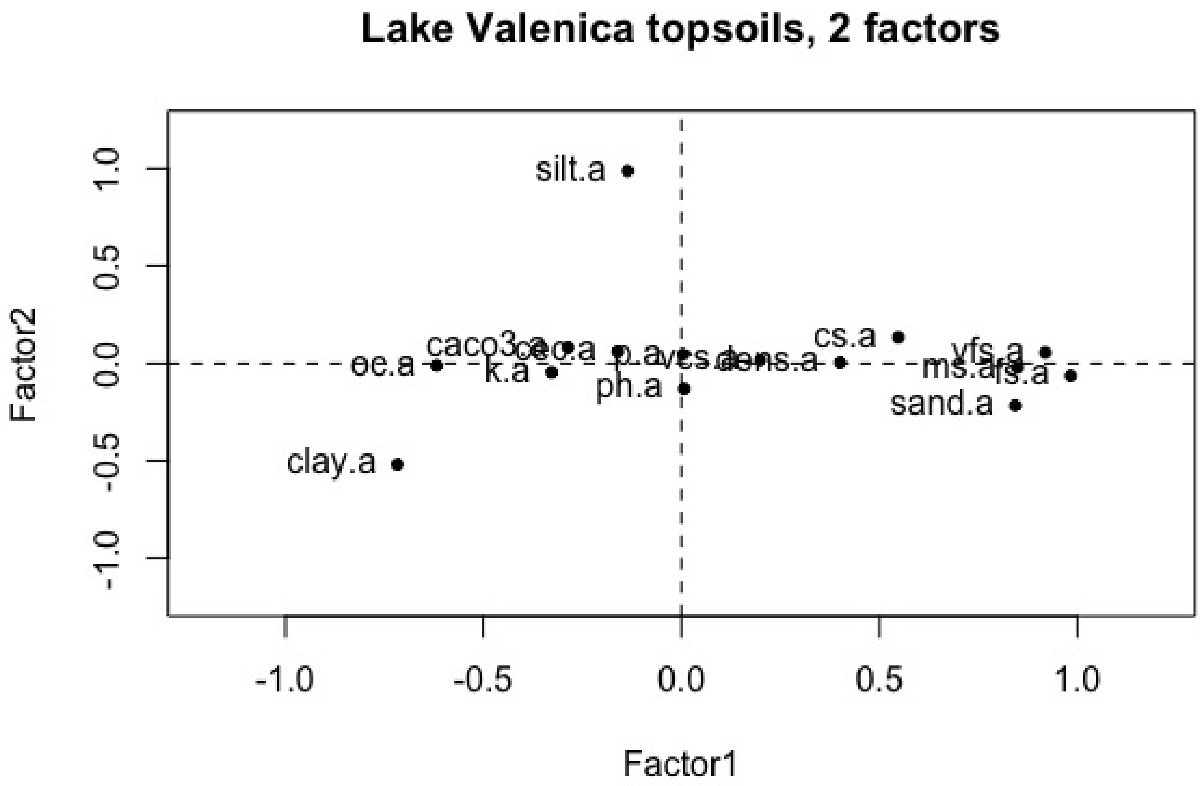
Hypothesize two latent variables (1) particle-size distribution (texture), (2) reaction (pH, carbonates)

```
> (fa <- factanal(dlv.r[4:18], 2))
Uniquenesses:
 dens.a   vcs.a    cs.a    ms.a    fs.a   vfs.a  sand.a  silt.a  clay.a
  0.840   0.960   0.683   0.280   0.030   0.153   0.243   0.005   0.218
   oc.a    ph.a   cec.a caco3.a     p.a     k.a
  0.618   0.983   0.970   0.911   0.998   0.891

               Factor1 Factor2
Cumulative Var   0.325   0.414

Loadings:
        Factor1 Factor2                 Factor1 Factor2                 Factor1 Factor2
dens.a   0.401                  vfs.a    0.919                  ph.a            -0.131
vcs.a    0.199                  sand.a   0.843  -0.216          cec.a   -0.161
cs.a     0.547   0.133          silt.a  -0.136   0.988          caco3.a -0.286
ms.a     0.848                  clay.a  -0.717  -0.519          p.a
fs.a     0.983                  oc.a    -0.618                  k.a     -0.328
```

Latent variables – interpretation

- **uniqueness**: Ψ, noise left over after factors are fitted, i.e., variable is unexplained
  - here P, K, pH, CaCO3 most unique → more factors are needed
- **variance explained** as in PCA. Note in PCA $k = p$ and all variance is eventually explained
- **loadings** as in PCA
  - Factor 1 is associated with all sand fractions (coarse-textured soils) and bulk density, opposed to clay concentration and organic C
  - Factor 2 is associated with silt opposed to clay
  - So both latent variables are most associated with particle-size distribution; *not* according to our original hypothesis – this should be modified

Latent variables – plot



Factor 1 (most variance explained) was rotated to show the maximum contrast (`varimax` rotation)

Latent variables – DTD images example

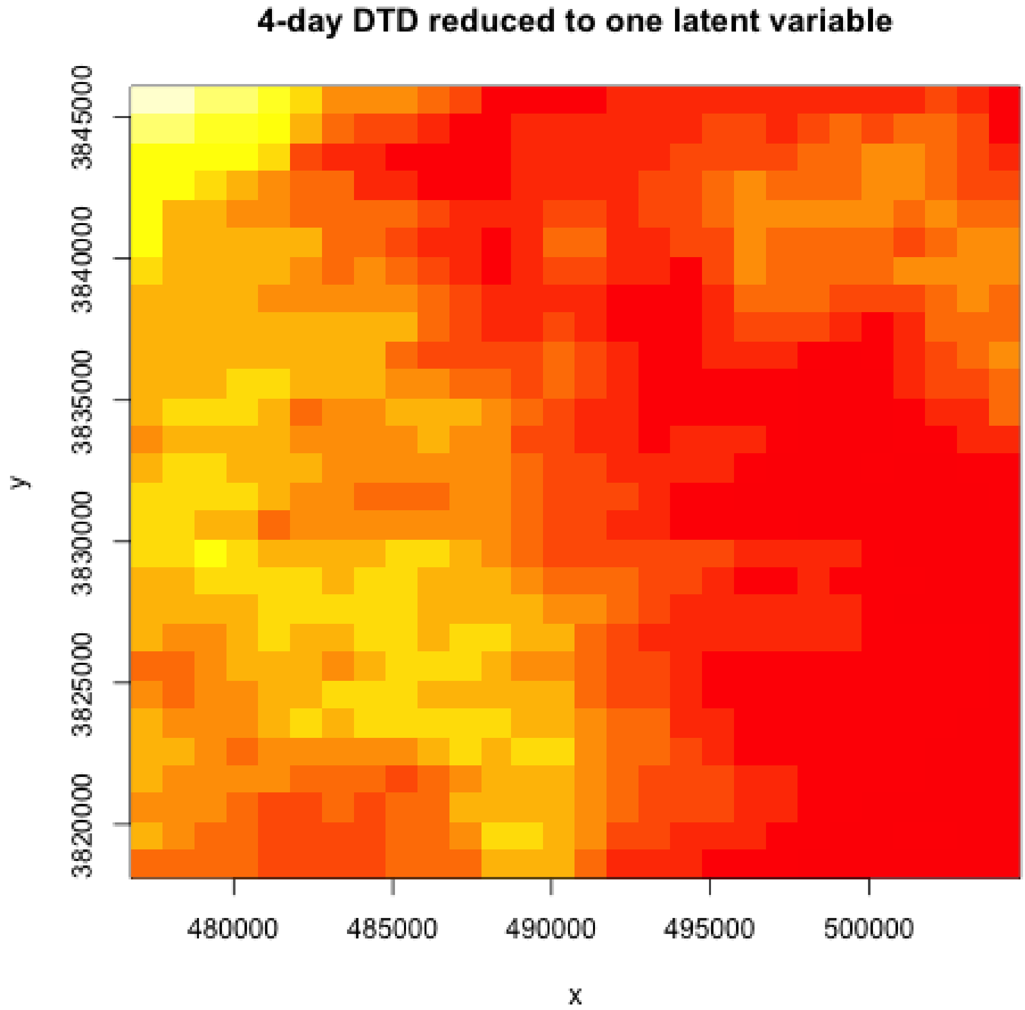
Hypothesis: one factor, "intensity"

```
Uniquenesses:
CK_DTD_304 CK_DTD_306 CK_DTD_307 CK_DTD_308
     0.181      0.047      0.073      0.171

Loadings:
           Factor1
CK_DTD_304 0.905
CK_DTD_306 0.976
CK_DTD_307 0.963
CK_DTD_308 0.910

               Factor1
SS loadings      3.527
Proportion Var   0.882
```

Strong correlation with each image, little uniqueness, 88% of variance explained.

Compare with PCA results.

Cornell University
College of Agriculture and Life Sciences



Single “best” image under the hypothesis of one process
Compare with PCA synthetic band 1.

Conclusion: PCA vs. latent variable analysis

PCA pure data reduction, maybe can interpret

Latent variable analysis aim is to produce interpretable variables representing the latent process

Cornell University
College of Agriculture and Life Sciences