Y-Aware Principal Components Regression in R

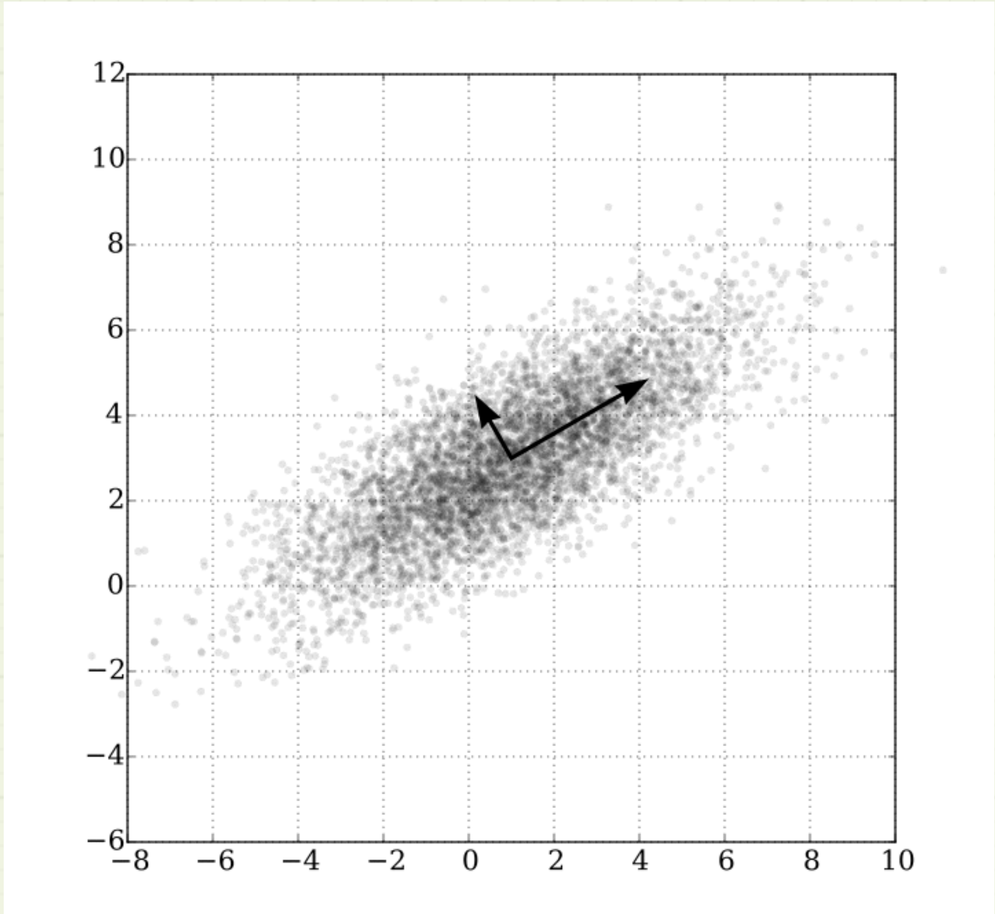
Nina Zumel
Win-Vector, LLC
August 9, 2016

Outline

- What is Principal Components Regression (PCR)?
- Some problems with the standard method
- Y-Aware PCR

What is Principal Components Regression (PCR)?

- Principal Components Analysis (PCA) + Regression
- Decompose *x* variables into orthogonal components
- Use a subset of the components to predict *y*
    - Usually: highest variance components (largest singular values)



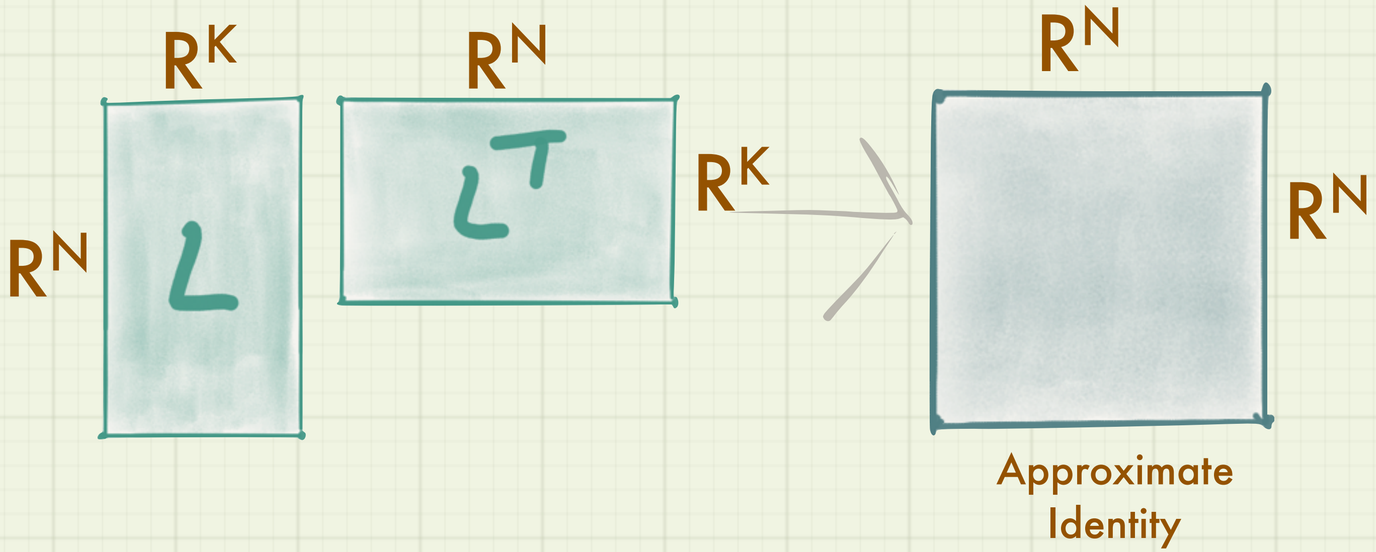
When is it useful?

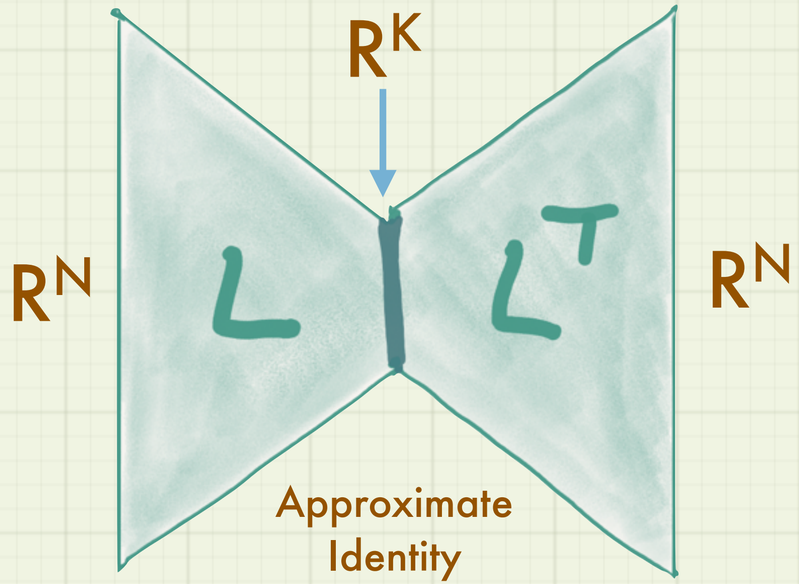
- You believe *y* depends on distinct latent processes
- You want to map observables (***x***) to those processes
- Dimensionality reduction
    - Especially with highly collinear variables

Example Applications

- Microarray data
  - More variables than observations
- Climate Analysis (Dendroecology)
  - Highly correlated/multicollinear variables (climate variables)
  - Model coefficients are important (infer climate vs. predict tree growth)
- Psychometrics (IQ)
  - "Intelligence" can't be directly measured
  - Function of unobservable latent factors
  - Observables: performance at various skills

Side-note: PCA as Linear Auto-encoding





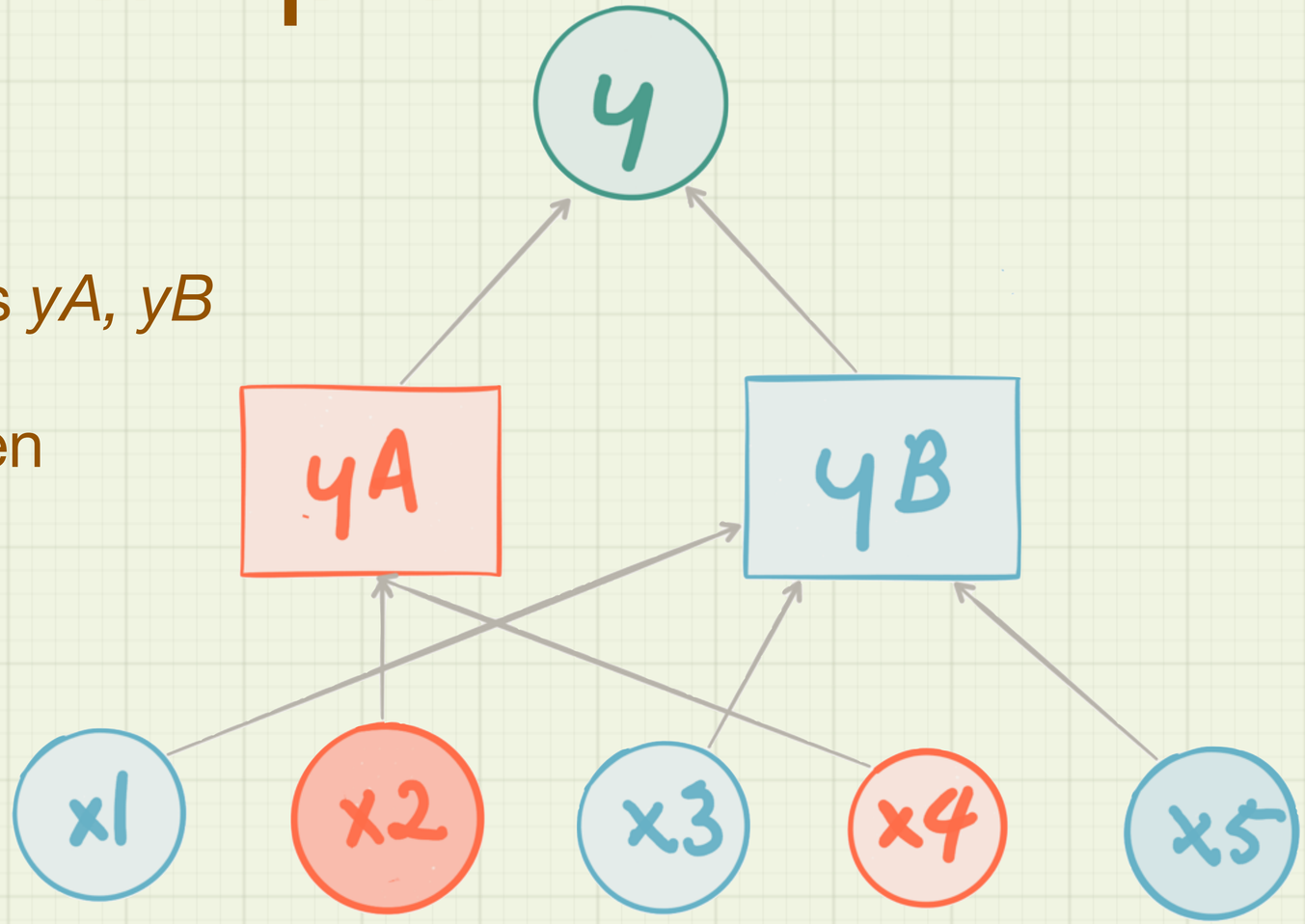
$K << N$

$(LL^T - I)x_i \sim 0$

LL^T defines smaller feature space in R^K

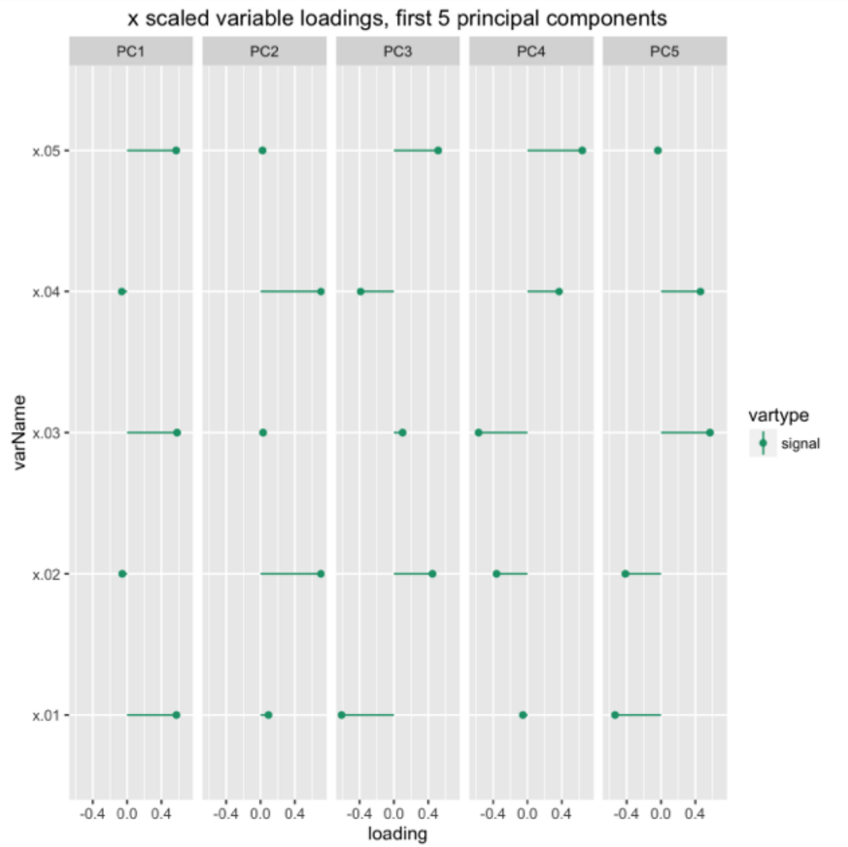
Example

- *y* is a sum of latent processes *yA, yB*
- Odd variables map to *yB*, even variables to *yA*
- Only *y* and x_i are observable
  - and noisy

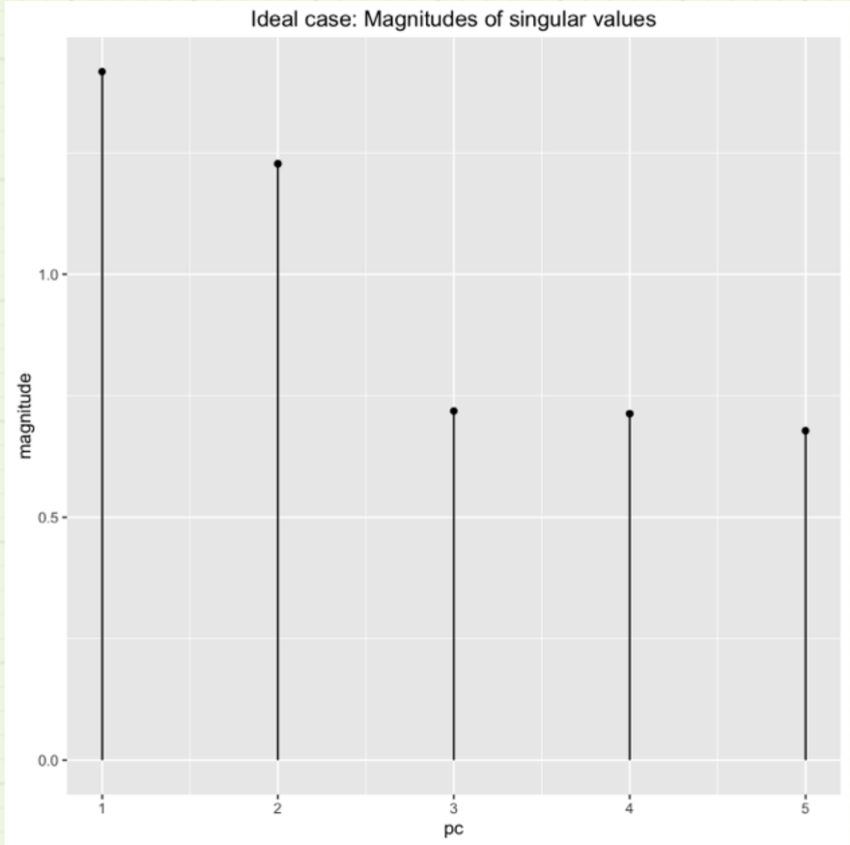


After PCA (Ideally)

```
princ <- prcomp(X,center = TRUE,scale. = TRUE)
```



`princ$rotation`

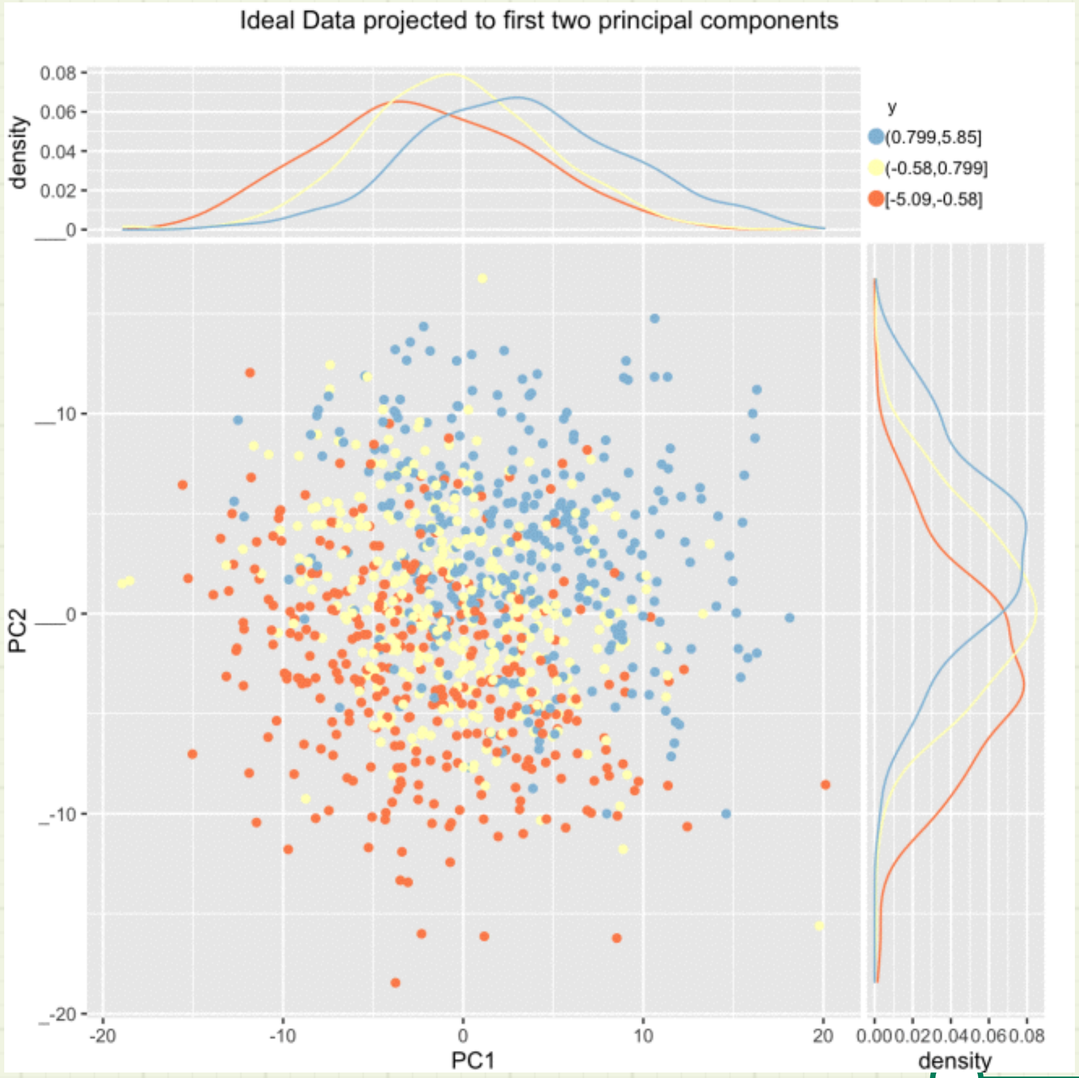


`princ$sdev`

From 5 variables to 2

```
proj <- X %*% princ$rotation[,1:2]
```

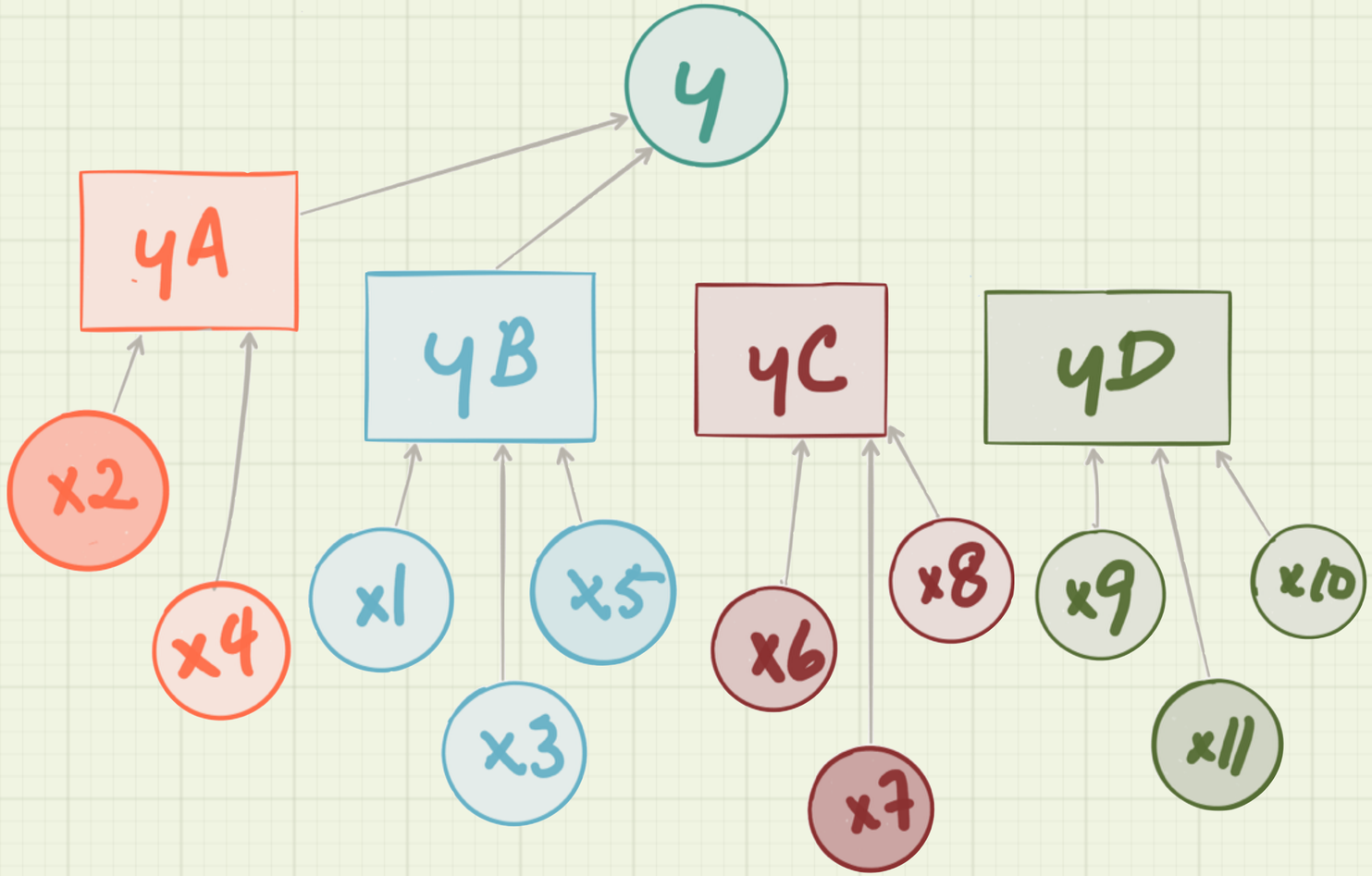
- As *PC1, PC2* increase, *y* increases
- *PC1* and *PC2* have captured the latent processes (*yA, yB*)



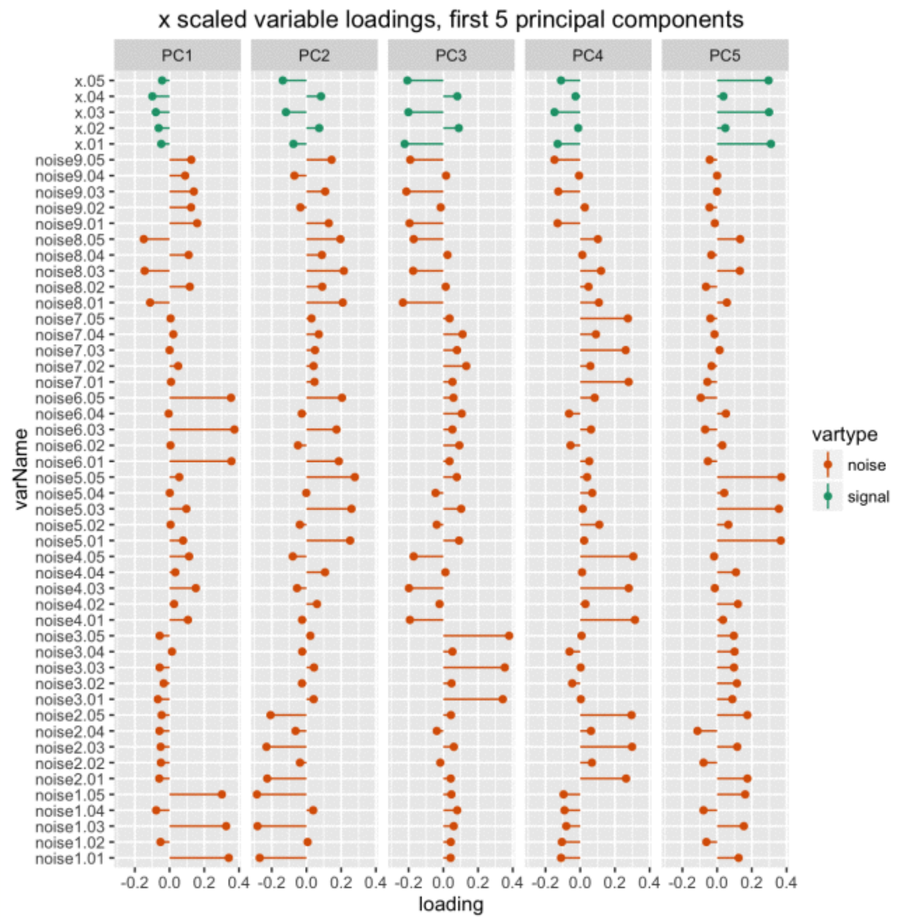
Win-Vector LLC

What can go wrong?

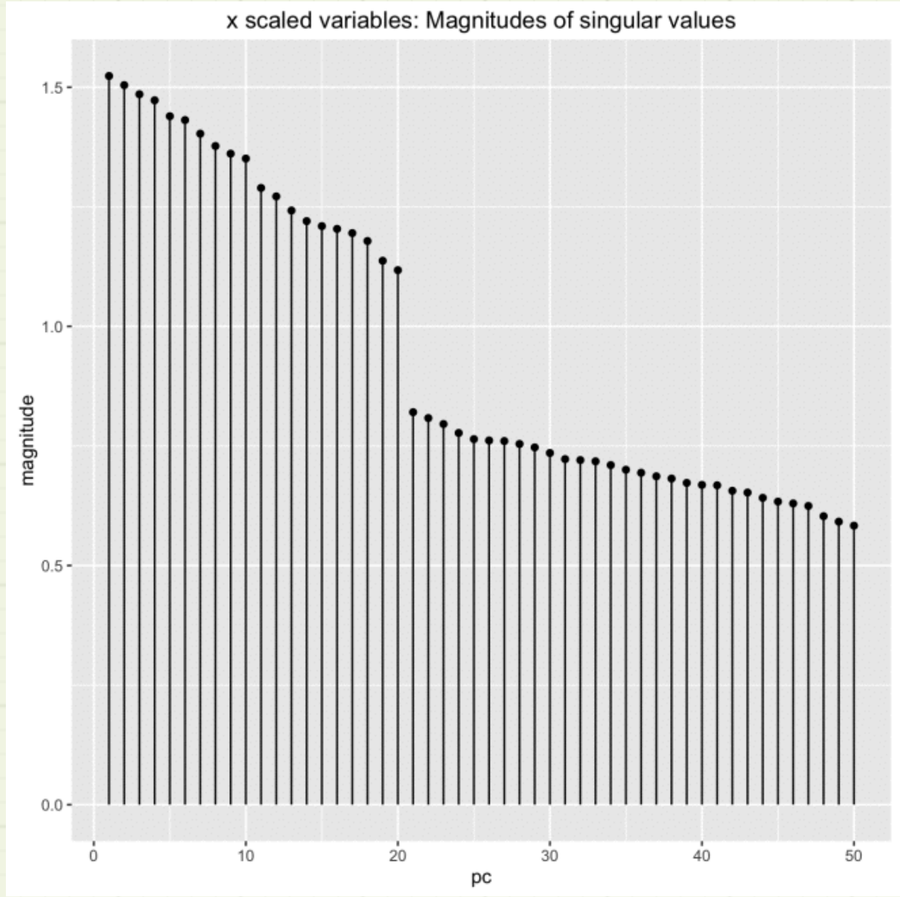
- Additional latent processes (correlations), NOT related to outcome of interest *y*
- Standard PCA tries to capture ALL latent structure



Resulting PCA



`princ$rotation[, 1:5]`



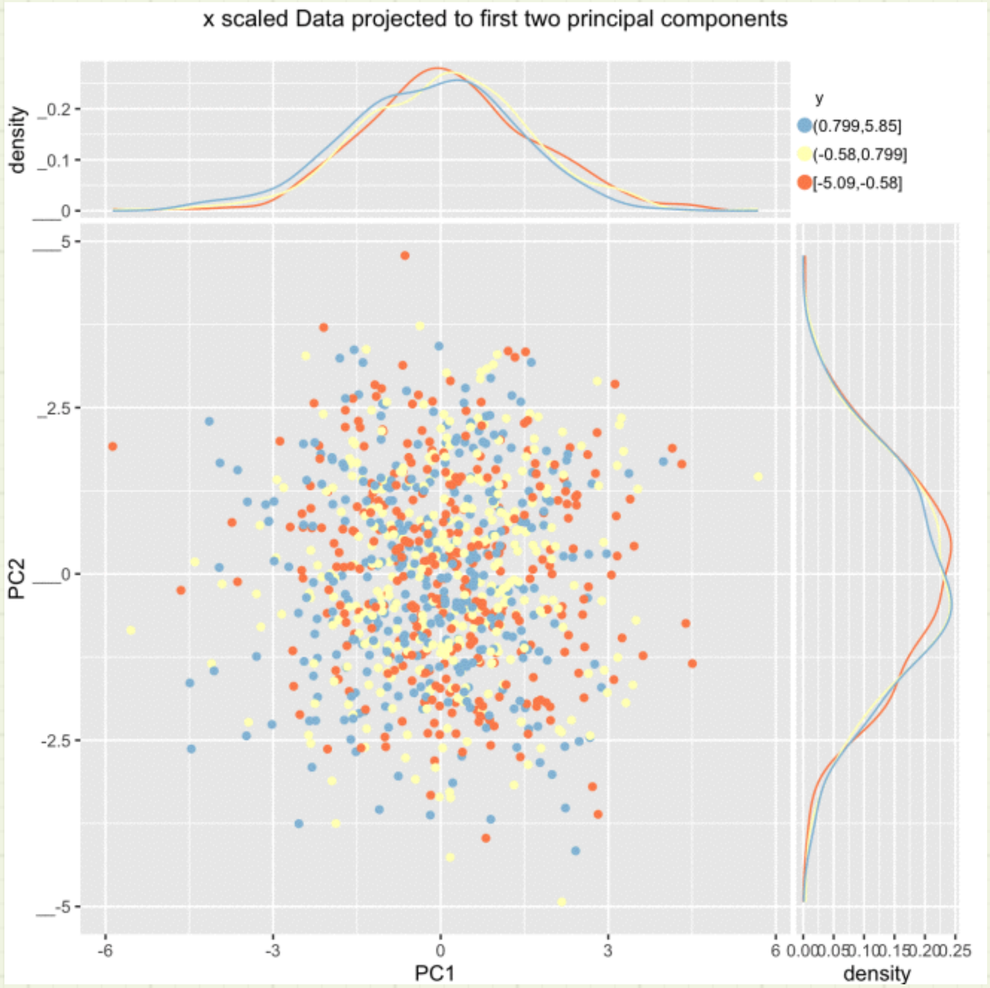
`princ$sdev`

Win-Vector LLC

Not enough to filter on High variance components

Regression on 20 vars: $R^2 = 0.48$

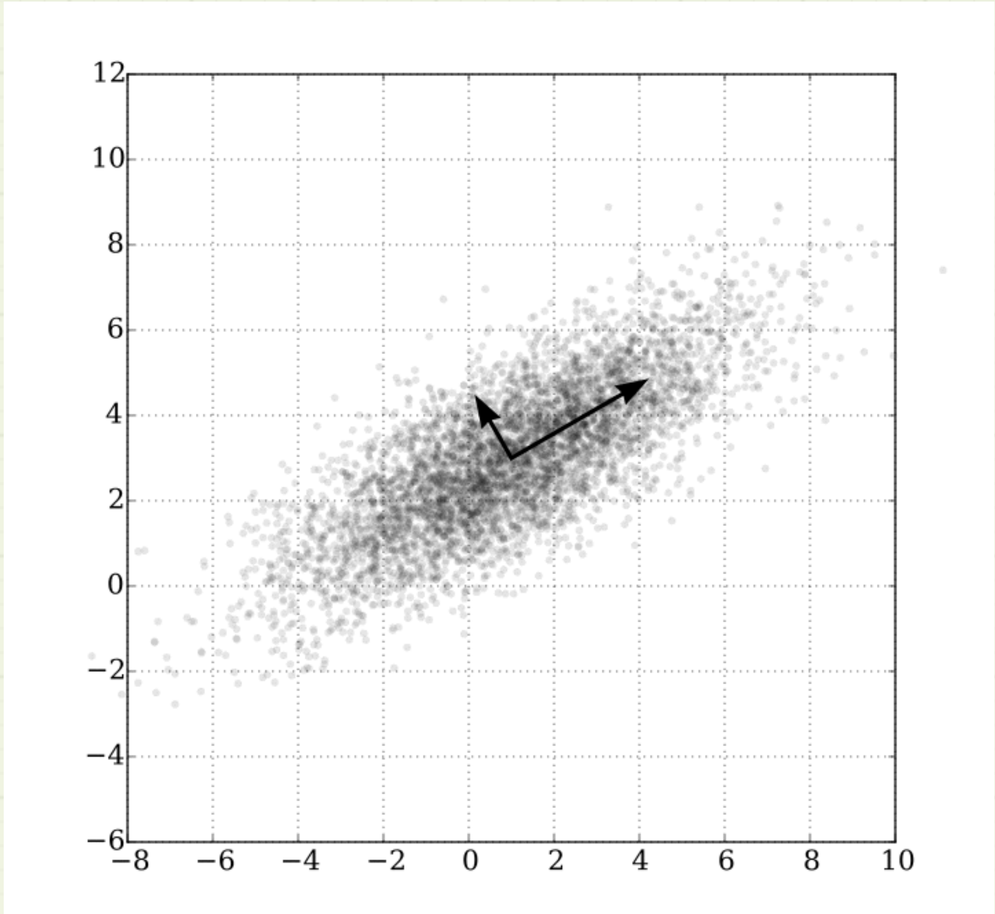
```
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.085043   0.039391   2.159 0.031097 *
## PC1          0.107016   0.025869   4.137 3.82e-05 ***
## PC2         -0.047934   0.026198  -1.830 0.067597 .
## PC3          0.135933   0.026534   5.123 3.62e-07 ***
## PC4         -0.162336   0.026761  -6.066 1.87e-09 ***
## PC5          0.356880   0.027381  13.034  < 2e-16 ***
## PC6         -0.126491   0.027534  -4.594 4.92e-06 ***
## PC7          0.092546   0.028093   3.294 0.001022 **
## PC8         -0.134252   0.028619  -4.691 3.11e-06 ***
## PC9          0.280126   0.028956   9.674  < 2e-16 ***
## PC10        -0.112623   0.029174  -3.860 0.000121 ***
## PC11        -0.065812   0.030564  -2.153 0.031542 *
## PC12         0.339129   0.030989  10.943  < 2e-16 ***
## PC13        -0.006817   0.031727  -0.215 0.829918
## PC14         0.086316   0.032302   2.672 0.007661 **
## PC15        -0.064822   0.032582  -1.989 0.046926 *
## PC16         0.300566   0.032739   9.181  < 2e-16 ***
## PC17        -0.339827   0.032979 -10.304  < 2e-16 ***
## PC18        -0.287752   0.033443  -8.604  < 2e-16 ***
## PC19         0.297290   0.034657   8.578  < 2e-16 ***
## PC20         0.084198   0.035265   2.388 0.017149 *
```



Win-Vector LLC

"Aha!" Moment

- Standard PCA: "x-space"
  - x' = (x - mean(x))/sd(x)
- But predictions are in "y-space"
  - So we should measure distances there



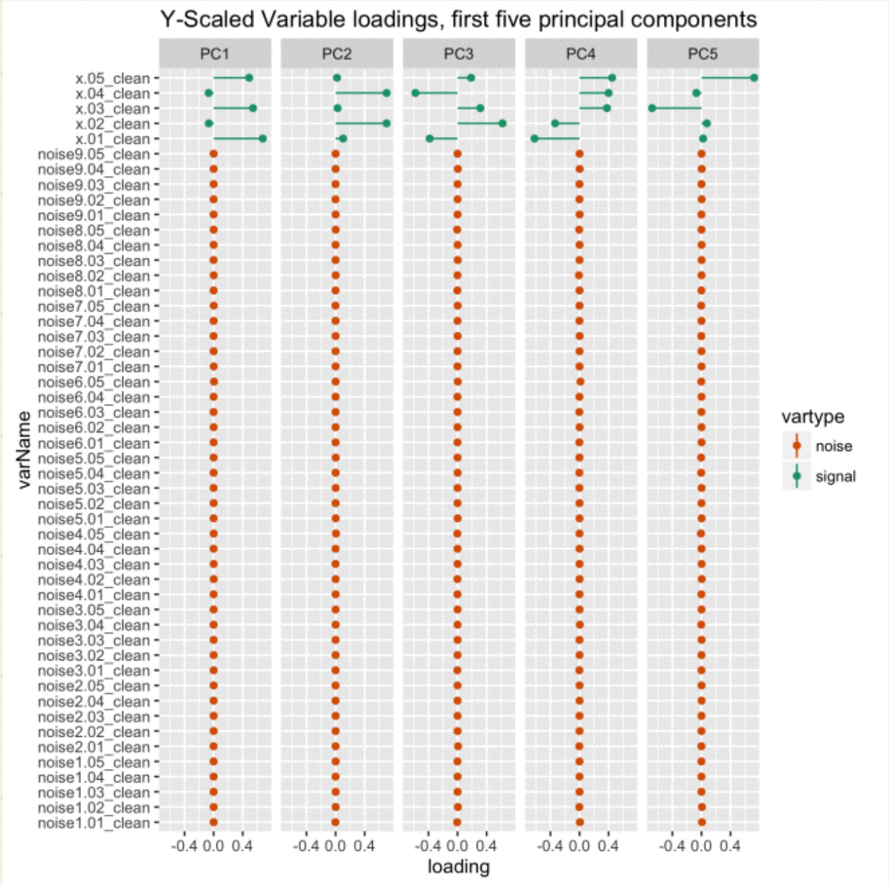
Y-Aware Scaling

Rescale variables to "y units"

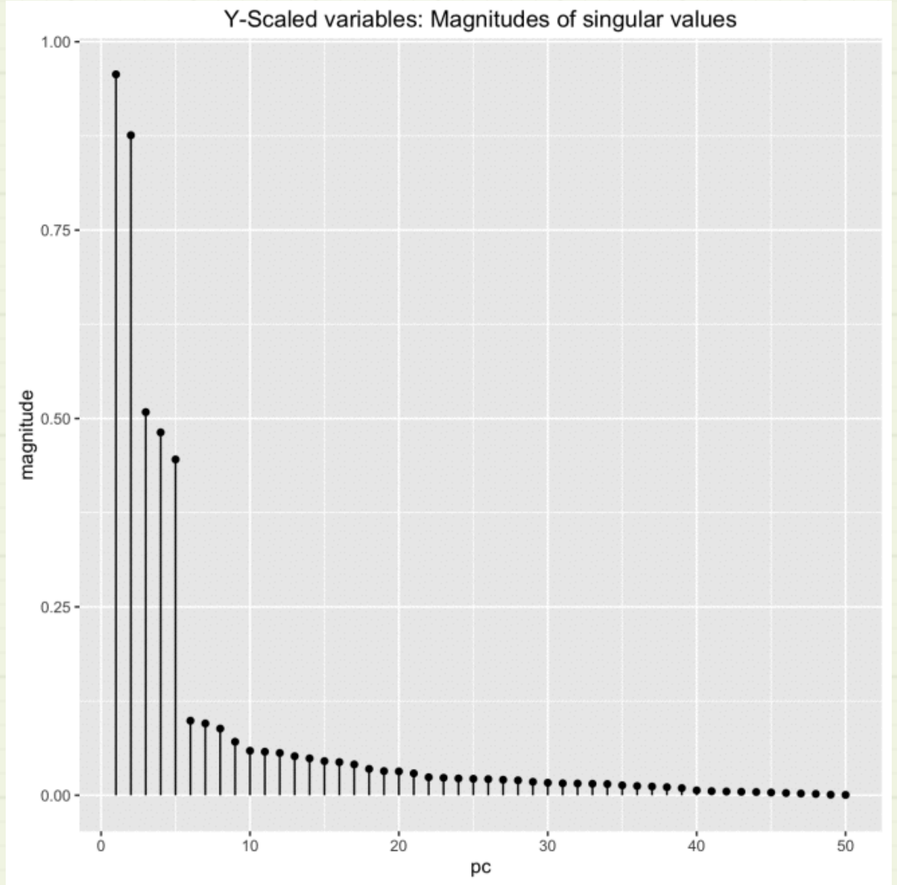
- Linear fit: $y = mx + b$
    - Unit change in $x \Rightarrow m$ units change in y (on average)
- Rescale: $x' = m(x - \text{mean}(x))$
    - Unit change in $x' \Rightarrow$ unit change in y
- (May want to center/scale y first: $y' = (y - \text{mean}(y))/\text{sdev}(y)$)
- One of the services in the `vtreat` package (available on CRAN)

Y-aware PCA

```
princ <- prcomp(Xyscaled,center = FALSE,scale. = FALSE)
```




```
prcomp$rotation[, 1:5]
```

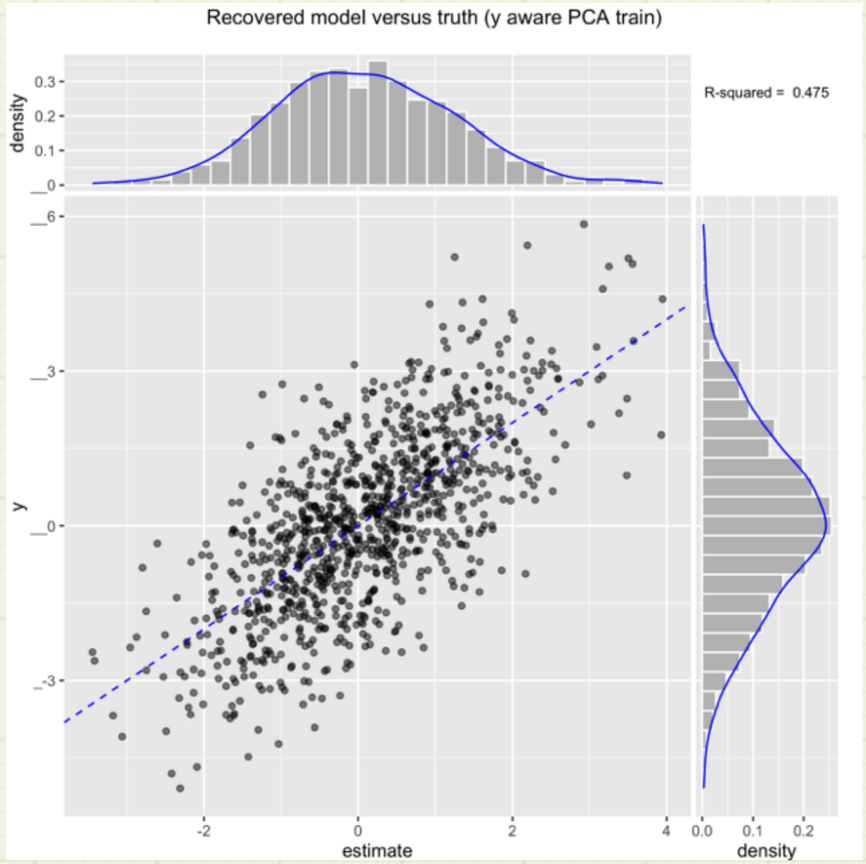


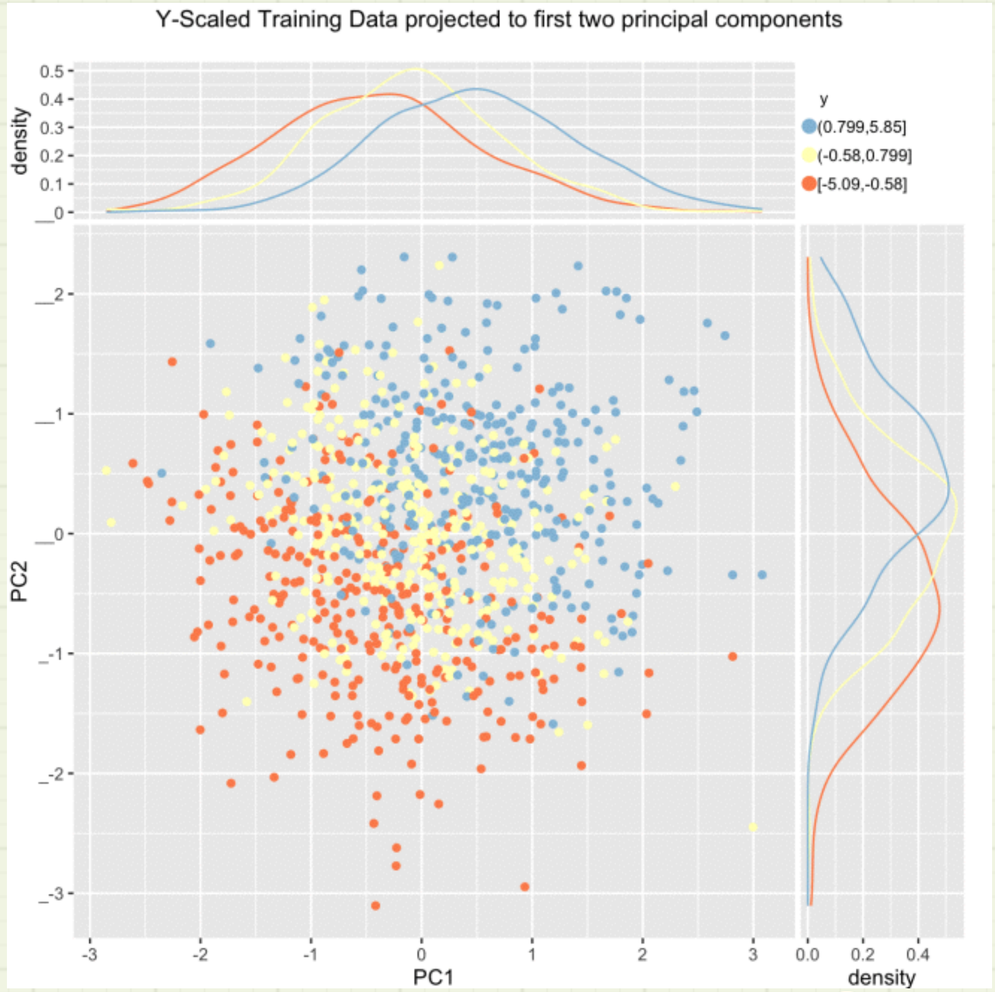

```
prcomp$sdev
```

Win-Vector LLC

Y-Aware PCR

Regression on 2 vars: $R^2 = 0.48$





Win-Vector LLC

Picking the Right Number of Components

- Standard (x-only) methods can work with modification
  - Jackson, Donald A. "Stopping Rules in Principal Components Analysis: A Comparison of Heuristical and Statistical Approaches", *Ecology* Vol 74, no. 8, 1993.
  - http://www.win-vector.com/blog/2016/05/pcr_part3_pickk/
- Significance Pruning
  - Again take advantage of y

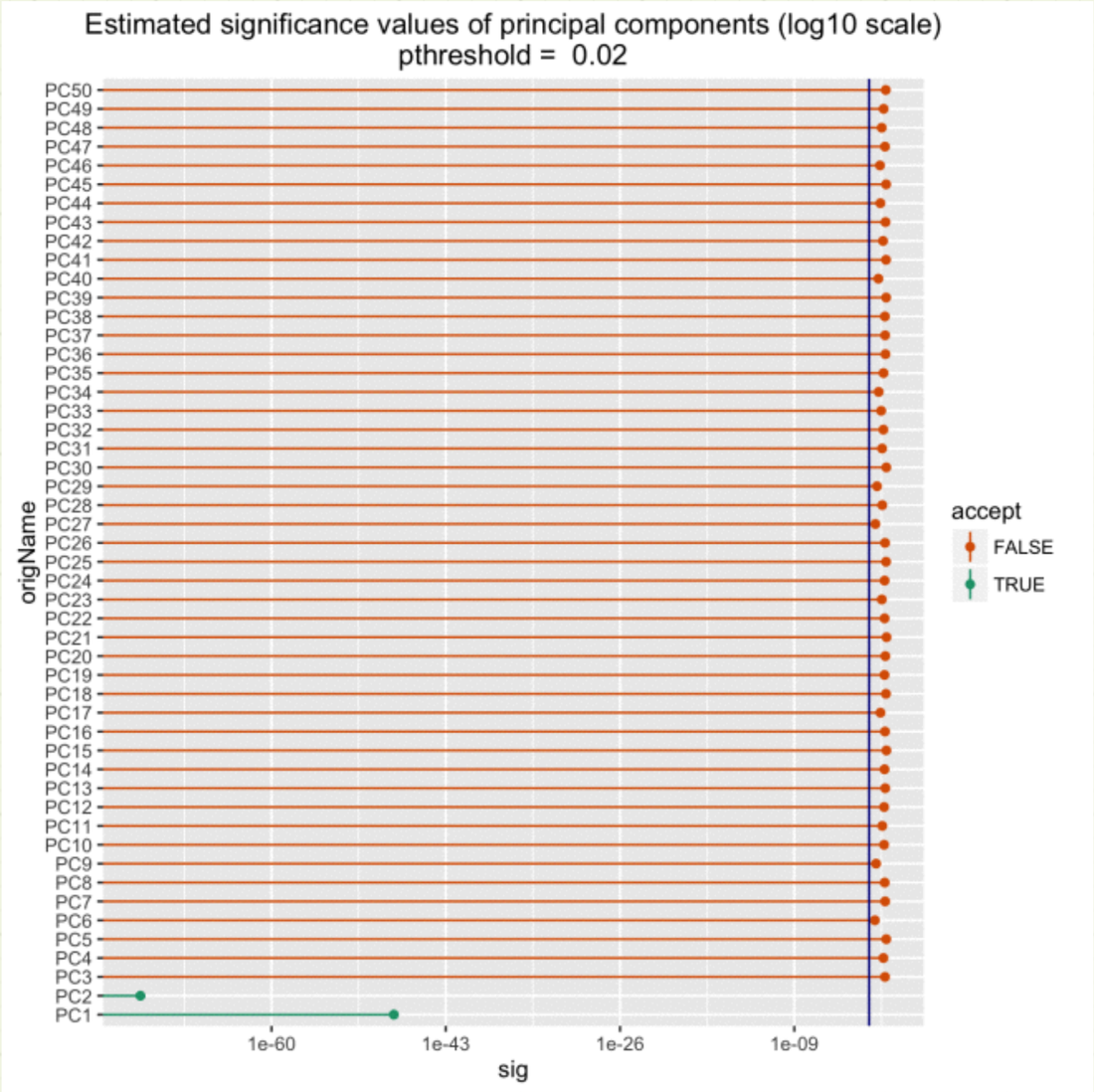
Significance Pruning

Significance of one-variable linear model

- $y = m\ PC + b$
- F = "explained variance"/"total variance"
  - distributed as the F distribution with appropriate degrees of freedom
  - http://facweb.cs.depaul.edu/sjost/csc423/documents/f-test-reg.htm
- Stop at first component to fail significance threshold
- http://www.win-vector.com/blog/2015/08/how-do-you-know-if-your-data-has-signal/

Significances for our Example

Accept variables
where
p-value of F < 0.02



Win-Vector LLC

Alternative Approaches

- Partial Least Squares (Wold *etal*, 1984)
- Supervised PCR (Bair *etal*, 2006)
  - http://web.stanford.edu/~hastie/Papers/spca_JASA.pdf
- Regularized Regression (Hastie, 2009)

Takeaways

- If you care about latent structure
    - Y-aware or Supervised PCR
- If you don't care about latent structure
    - Significance pruning or regularization
    - Or an alternative modeling approach
- You can also combine methods:
    - significance prune ⇒ Y-aware PCA ⇒ significance prune ⇒ regression
    - significance prune ⇒ Y-aware PCA ⇒ regularized regression

R Code

- https://github.com/WinVector/Examples/tree/master/PCR

Thank You