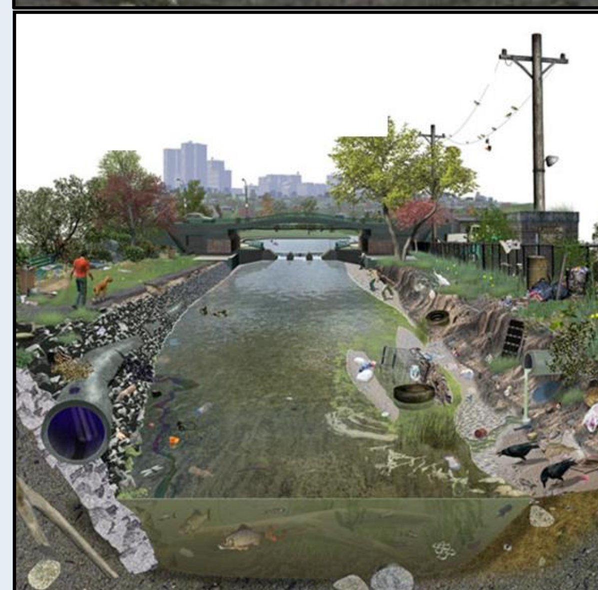
Small *n* and collinear predictors: assessment of alternative regression methods for LULC studies

Brad Schneid
Christopher Anderson
Jack Feminella

Auburn University

Background

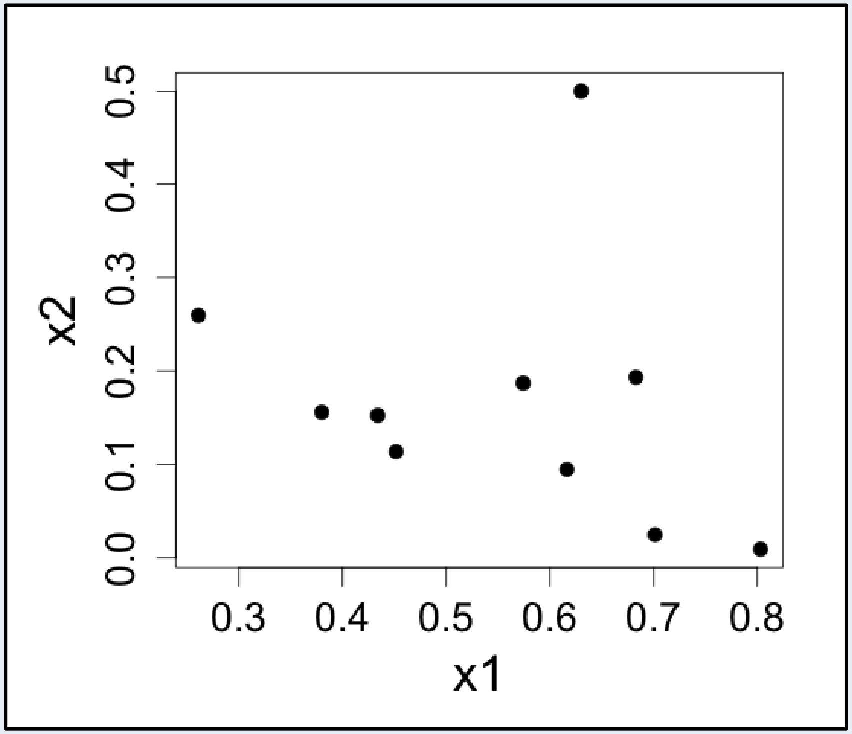


Agricultural & urban land-use/cover (**LULC**) can influence:

- stream hydrology
- nutrient & sediment concentrations
- aquatic habitat quantity/quality
- thermal regime
- aquatic biota

Background

- LULC impact studies frequently have:
  - Low replication (small n)
  - Collinear predictors
  - Outliers/non-ideal data



Background & Goals

- Ordinary least-squares (**OLS**) performs poorly with collinearity and/or small *n*

$$\widehat{var}\left(\hat{\beta}_j\right) = \frac{s^2}{(n-1)var(X_j)} \cdot \frac{1}{1-R_j^2}$$

- Compare OLS-related selection methods & alternatives in terms of:
  - Identification of "important" predictors
  - Coefficient estimation and prediction

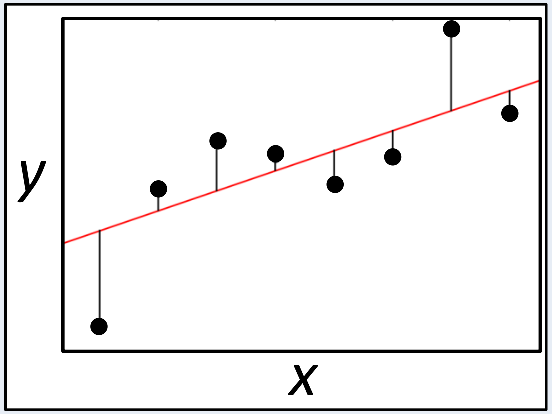
OLS selection methods

Many OLS-related selection methods/criteria exist

- Automated **stepwise** methods (forward, backward)
  - Criteria: adjusted R^2, Mallow's Cp, AIC, etc…
    - AIC (Akaike information criterion)
      - model fit (SS_{res}) + penalty for model size
      - allows for model ranking/weighting
      - **AICc** = small sample size corrected AIC
- Multi-model averaging (**MMA**)
  - 'natural' (MMA.n) and 'zero' (MMA.z) methods

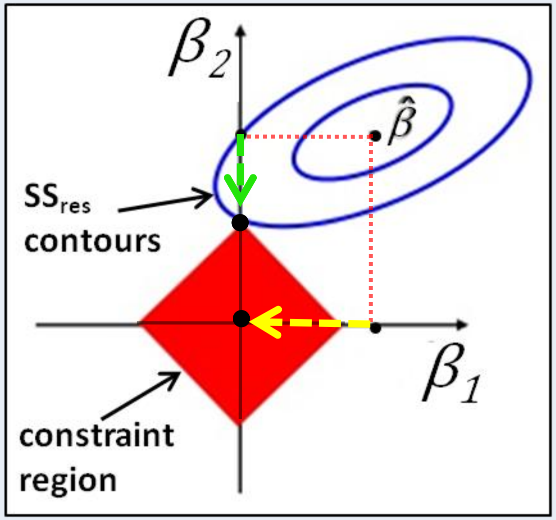
OLS and coefficient ‘shrinkage’

- **OLS** seeks $\hat{\beta}$ that minimizes the sum of squared residuals $(SS_{res}) = \Sigma(y - \mathbf{X}\beta)^2$



- **LASSO** (least absolute shrinkage & selection operator) modifies OLS solution to constrain the absolute magnitude of reg. coefficients:

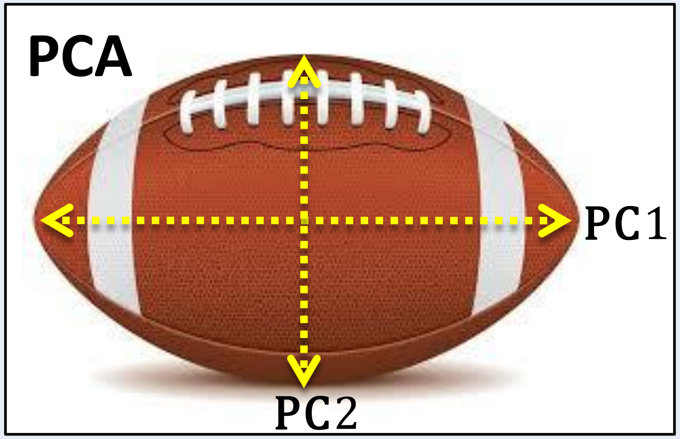
LASSO seeks $\hat{\beta}$ to min. SS_{res}, s.t. $\Sigma|\beta| \leq t$

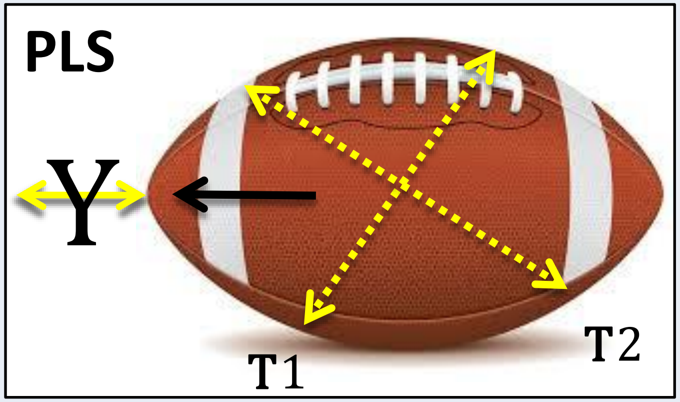


* t is a tuning parameter, chosen by C.V.

Latent variables: PLS vs. PCA

Football =
centered **X** cloud





PCA:

Find **PC**s = **X**W such that:

- *var*(PC1) > *var*(PC2) > … > *var*(PCp)
- PCi ⊥ PCj for all i,j pairs

PLS:

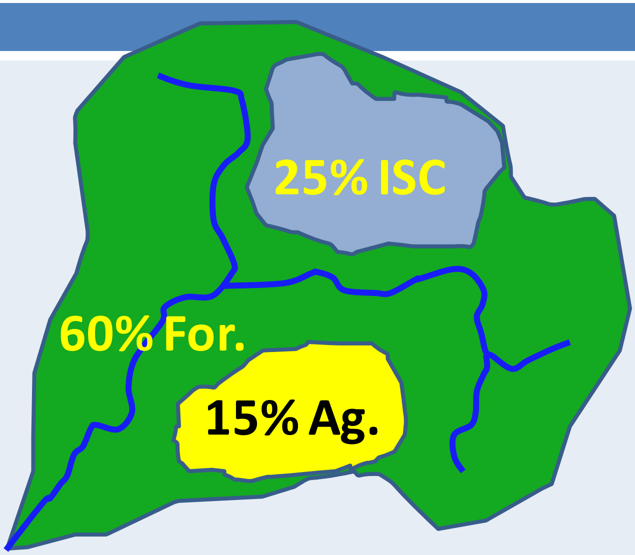
1) Find **T**s = **X**W such that:

- cov(Y, T1) > cov(Y, T2) > … > cov(Y, Tp)
- **T**i ⊥ **T**j for all i,j pairs

2) Regress Y on **T**, get coefficients (Q)

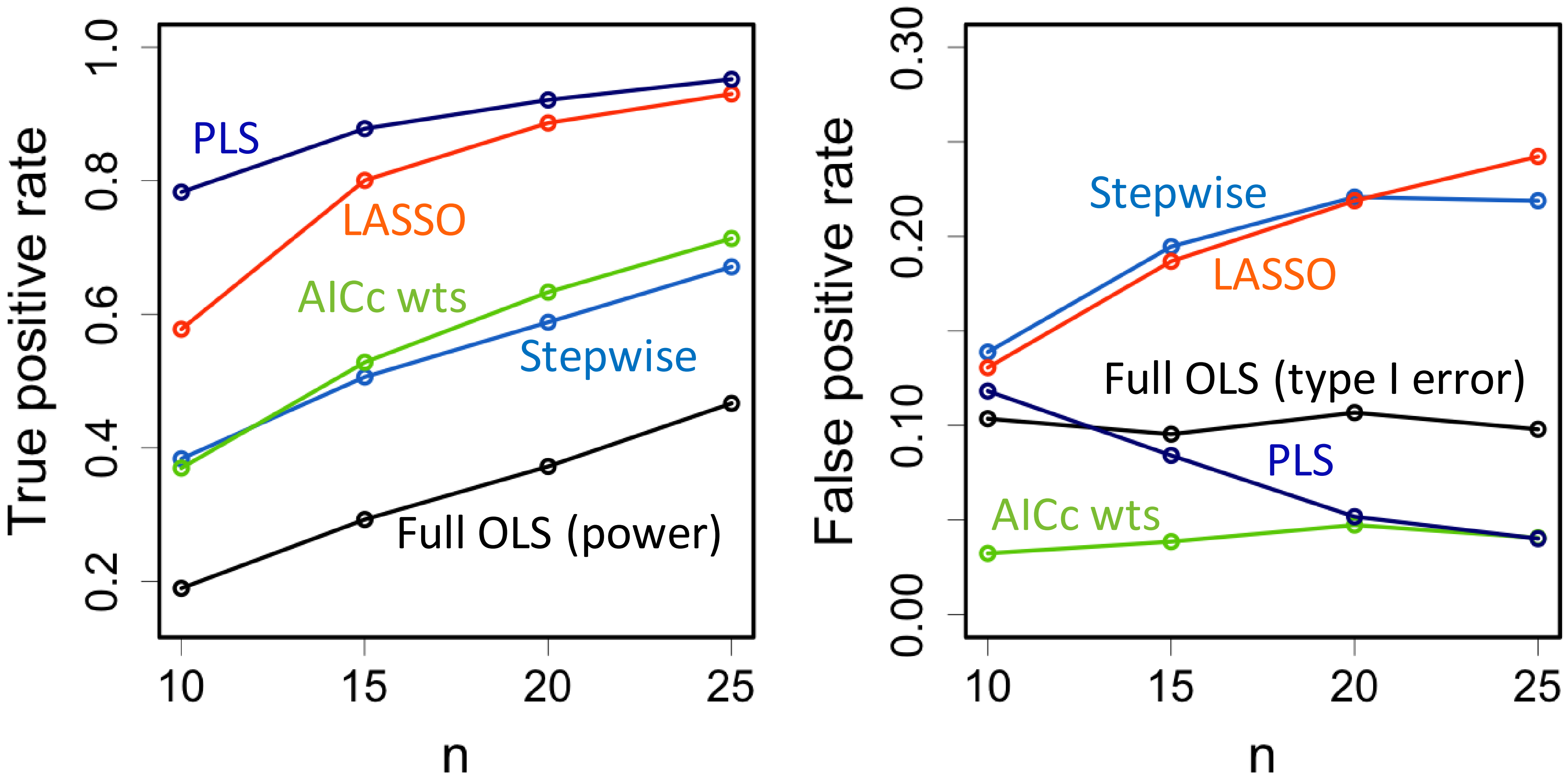
3) Calculate **X** coefficients: $\beta = WQ'$

Methods: data creation

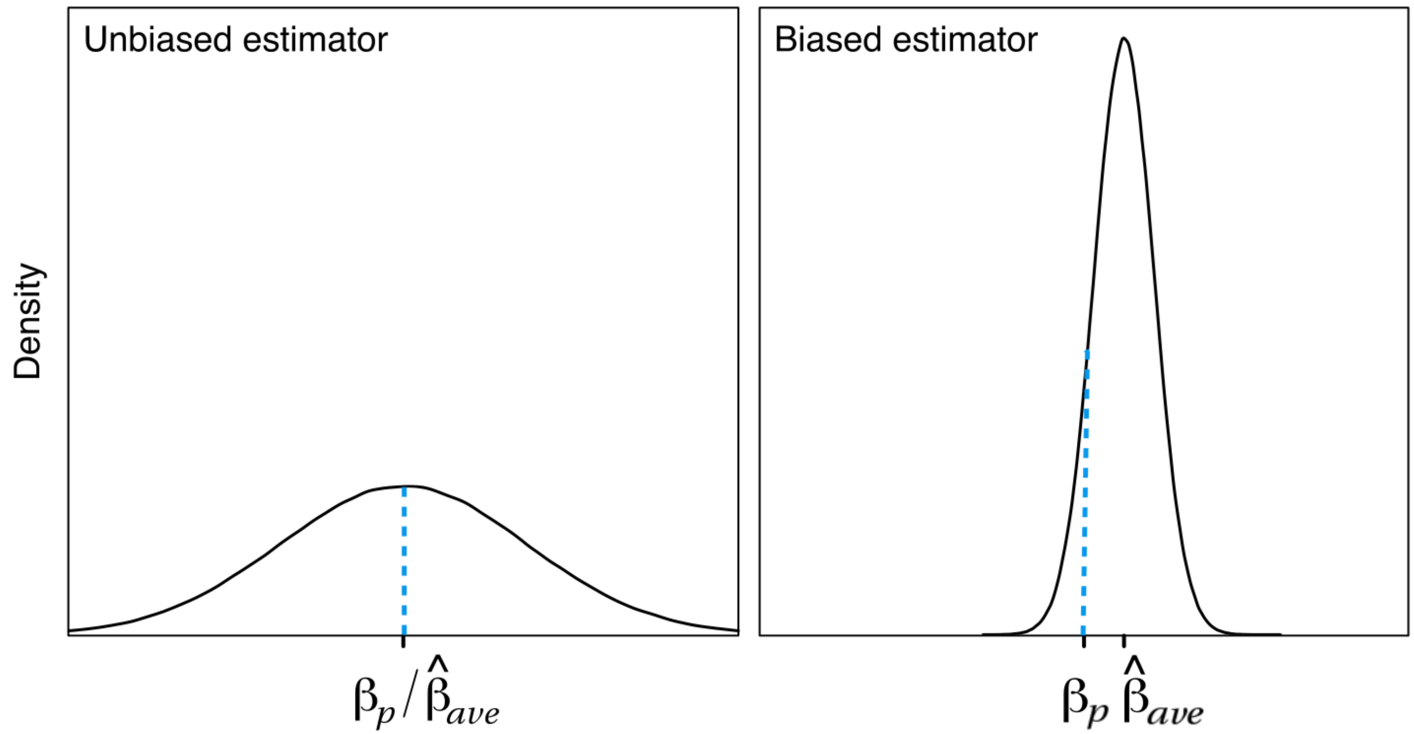


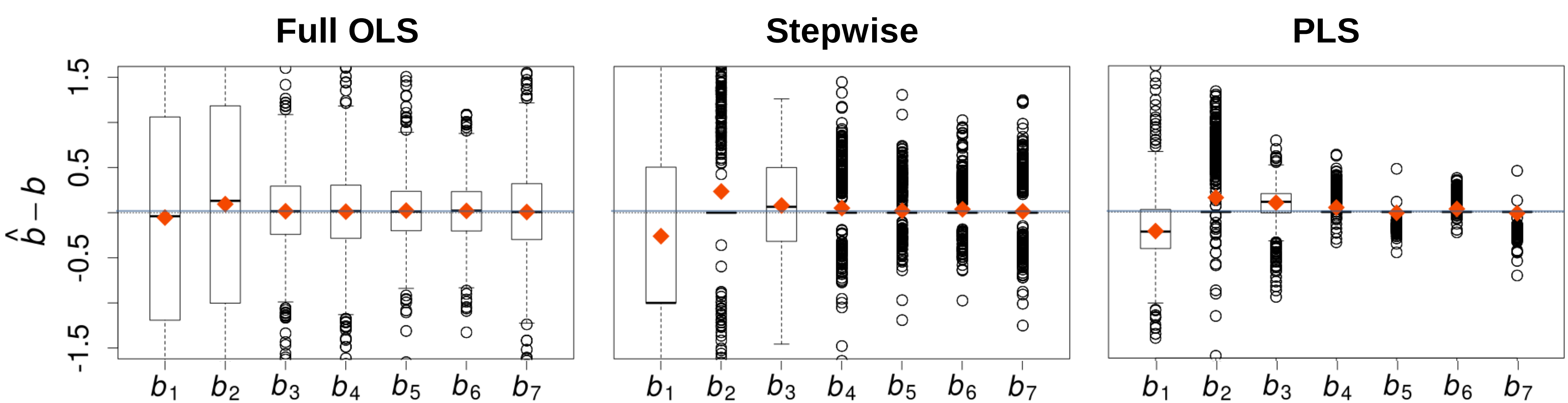
- Simulated $\mathbf{X}$ to mimic % LULC
  - 6 highly correlated X variables
    - average simulation correlation ≈ 0.70
  - Created 1 additional "forest" variable (7 total)
    - $X_3 = 1 - (X_1 + X_2 + \text{noise})$
  - Remaining Xs named X_4 to X_7
- Let $y = X_1\beta_1 + X_3\beta_3 + N(0, 0.1)$; $\beta_1 = 1.0$, $\beta_3 = -0.5$

Results: classification of 'important' predictors



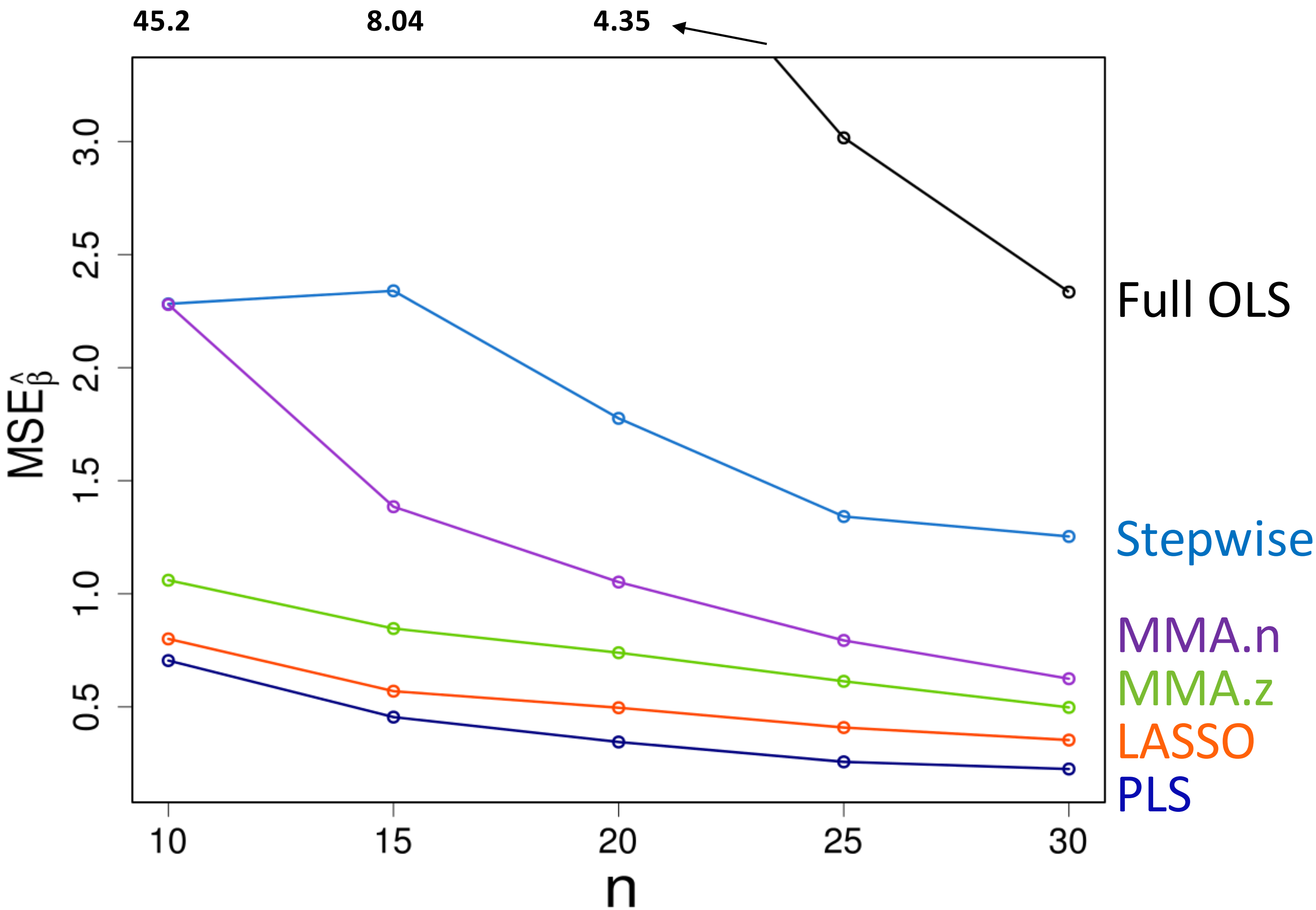
Bias-variance tradeoff



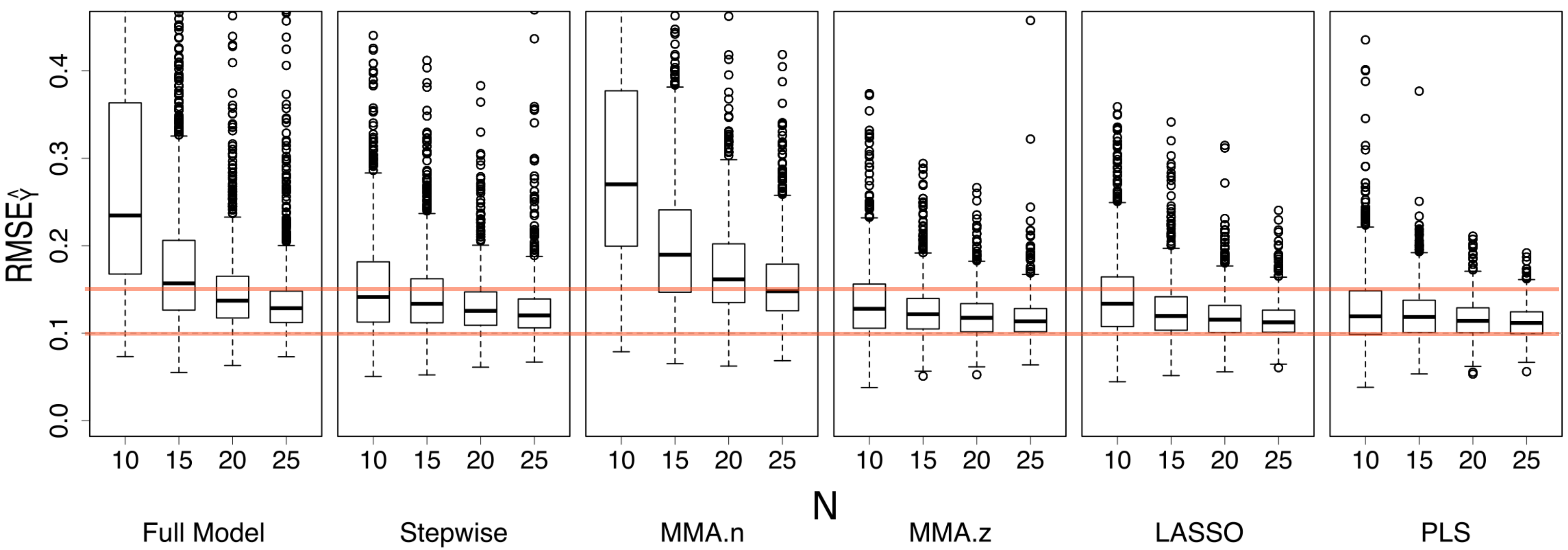


n = 15

Results: $MSE_{\hat{\beta}}$ = variance + bias^2



Results: prediction of test data



Conclusions & remarks

- PLS performed well with small *n* and highly collinear **X** with all 3 criteria
  - Identification of ‘important’ Xs
  - β estimation
  - Prediction
- PLS assumes that *y* is a function of underlying and unmeasured latent variables
- PLS also used for multiple Y regression/ordination

A simple modification for robust PLS

Brad Schneid and Ash Abebe

Background:

- PLS is sensitive to outliers, which are difficult to detect in multivariate data
- Outliers can be present as:
  1) entire incorrect observations (rows)
  2) recording/copying mistakes (random point values)
  3) correct, valid data

Background: PLS algorithms

- Available outlier-resistant PLS algorithms are overly complicated and tailored for entire outlying rows

- PLS algorithm begins by calculating covariance between **X** and *y* as initial weights

Goal: Determine if replacement of covariance with rank-based alternative results in outlier-resistant PLS

Methods: PLS modification

Pearson's ***r*** = *covariance*(*x*,*y*) / sd(*x*)sd(*y*)

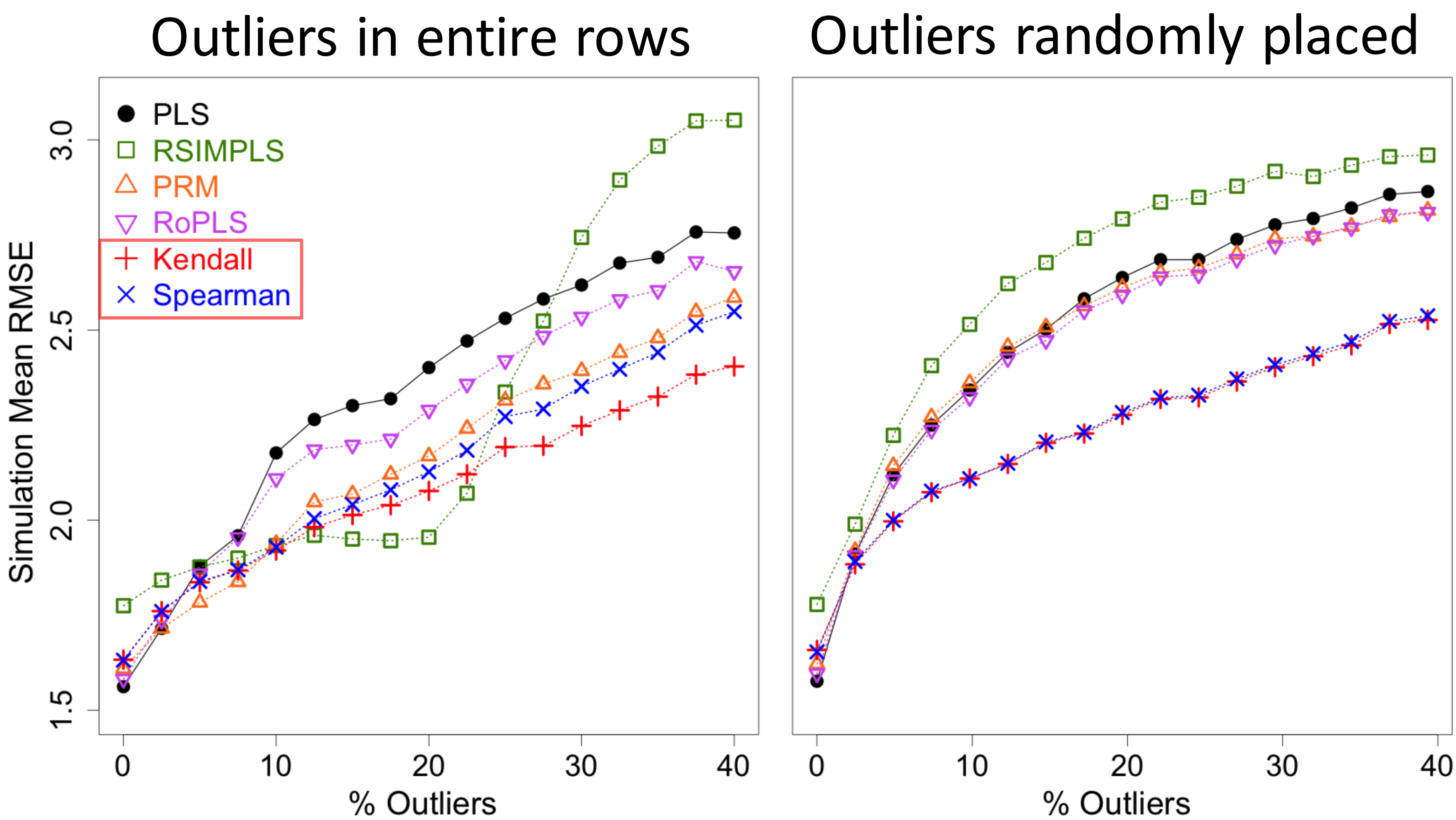
Robust covariance based on rank correlation:

- Spearman's ***rho***
- *Kendall's* ***tau***

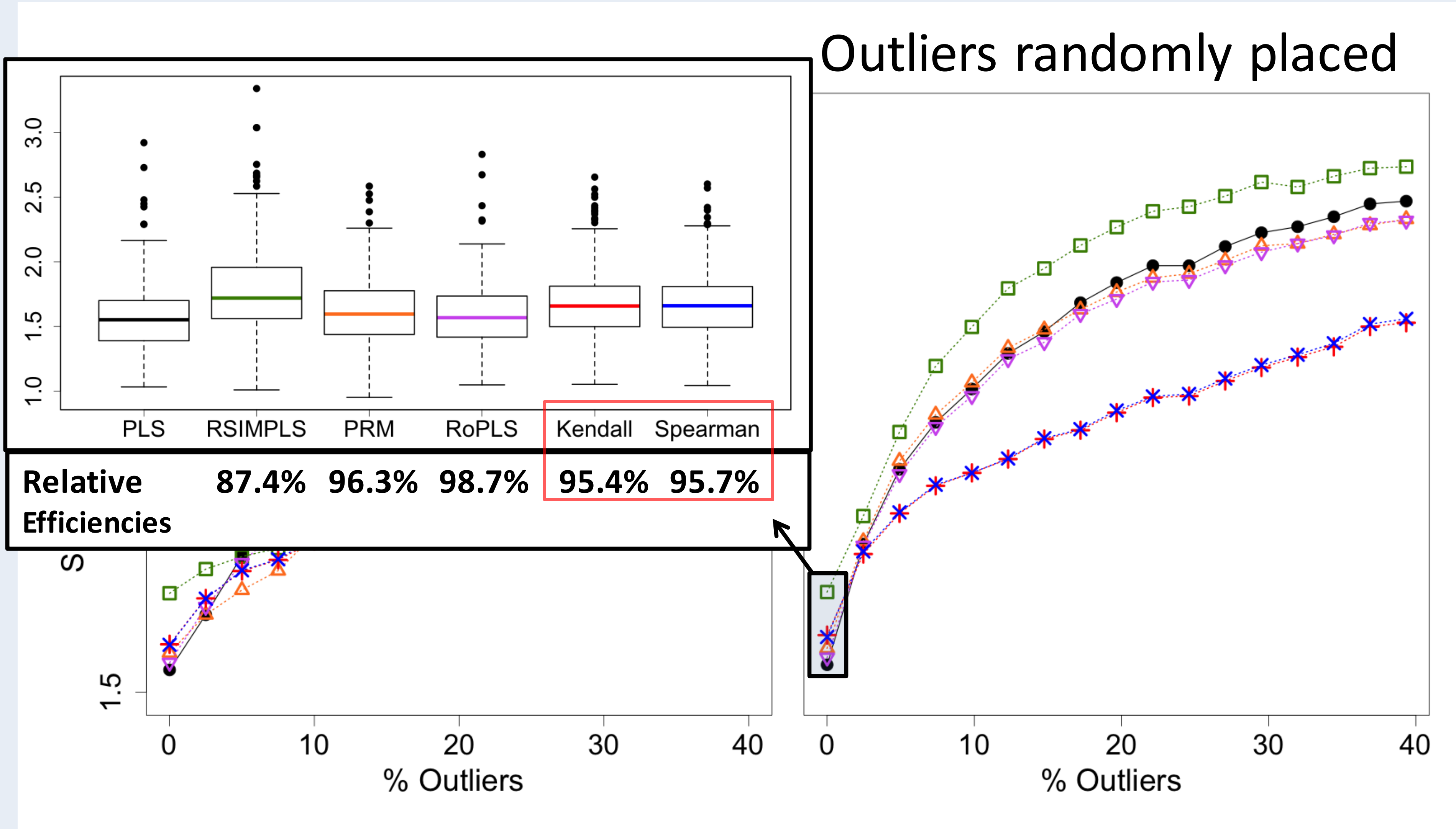
Angular transformation: 2sin([π(***rho***)]/6) *&* sin([π(***tau***)]/2)

Alt. covariance = transformed rank cor(*y*, *x*) • sd(*x*)sd(*y*)

Results: Prediction RMSE



Results: Prediction RMSE



Summary

Rank-based PLS:

- demonstrated outlier resistance in terms of β-est. and prediction in both outlier cases (rows & random)
- only methods resistant to randomly placed outliers
- had high relative efficiency

Questions?