Analyzing High-Dimensional Data

Challenges in analysis of ‘omics data

- New technologies now make it possible to capture huge amounts of data on experimental units.
  - **Genomics**
    - Can sequence and entire genome
    - Sequence all RNA in a tissues
  - **Phenomics**
    - Collect multi-spectral or hyper spectral data on plants throughout the growing season
- These technologies provide opportunities to increase understanding of complex phenotypes but can be challenging to model.
  - The ‘curse’ of dimensionality

R Exercise – N<<P Challenges

Methods for dealing with high-dimensional data

- **Penalized Methods** apply some penalty to the solutions to avoid overfitting the data when there are a large number of explanatory variables relative to the number of independent observations.
  - Various penalized methods differ in the type of penalty applied
  - Penalize methods yield solutions that are not unbiased – at least in the same sense that OLS solutions are unbiased

- **Dimension Reduction Methods** solve the issue of dimensionality be reducing the number of explanatory variables used to fit the regression.
  - Reducing the dimensions of the data inevitably results in the loss of some information

Penalized methods – Ridge Regression

- Ridge Regression simply adds some constant value to the diagonal of **X'X**

OLS solutions $$\hat{\mathbf{b}} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{y}$$

Ridge Regression Solutions solutions $$\hat{\mathbf{b}}^{*} = (\mathbf{X}'\mathbf{X} + \boldsymbol{\alpha}\mathbf{I})^{-1}\mathbf{X}'\mathbf{y}$$

Genomic Prediction using Mixed Models

$$\mathrm{u} = \sum w_i \beta_i \qquad \mathrm{W} = \mathrm{M} - \mathrm{P} \qquad \alpha = \frac{\sigma_e^2}{\sigma_\beta^2}$$

$$\begin{bmatrix} \mathrm{X}'\mathrm{X} & \mathrm{X}'\mathrm{W} \\ \mathrm{W}'\mathrm{X} & \mathrm{W}'\mathrm{W} + \alpha\mathrm{I} \end{bmatrix} \begin{bmatrix} \mathrm{b} \\ \beta \end{bmatrix} = \begin{bmatrix} \mathrm{X}'\mathrm{y} \\ \mathrm{W}'\mathrm{y} \end{bmatrix}$$

R – Exercise –Ridge Regression

Penalized Methods - LASSO

OLS minimizes the function: $(\mathrm{y} - \mathbf{X}\mathrm{b})'(\mathrm{y} - \mathbf{X}\mathrm{b})$

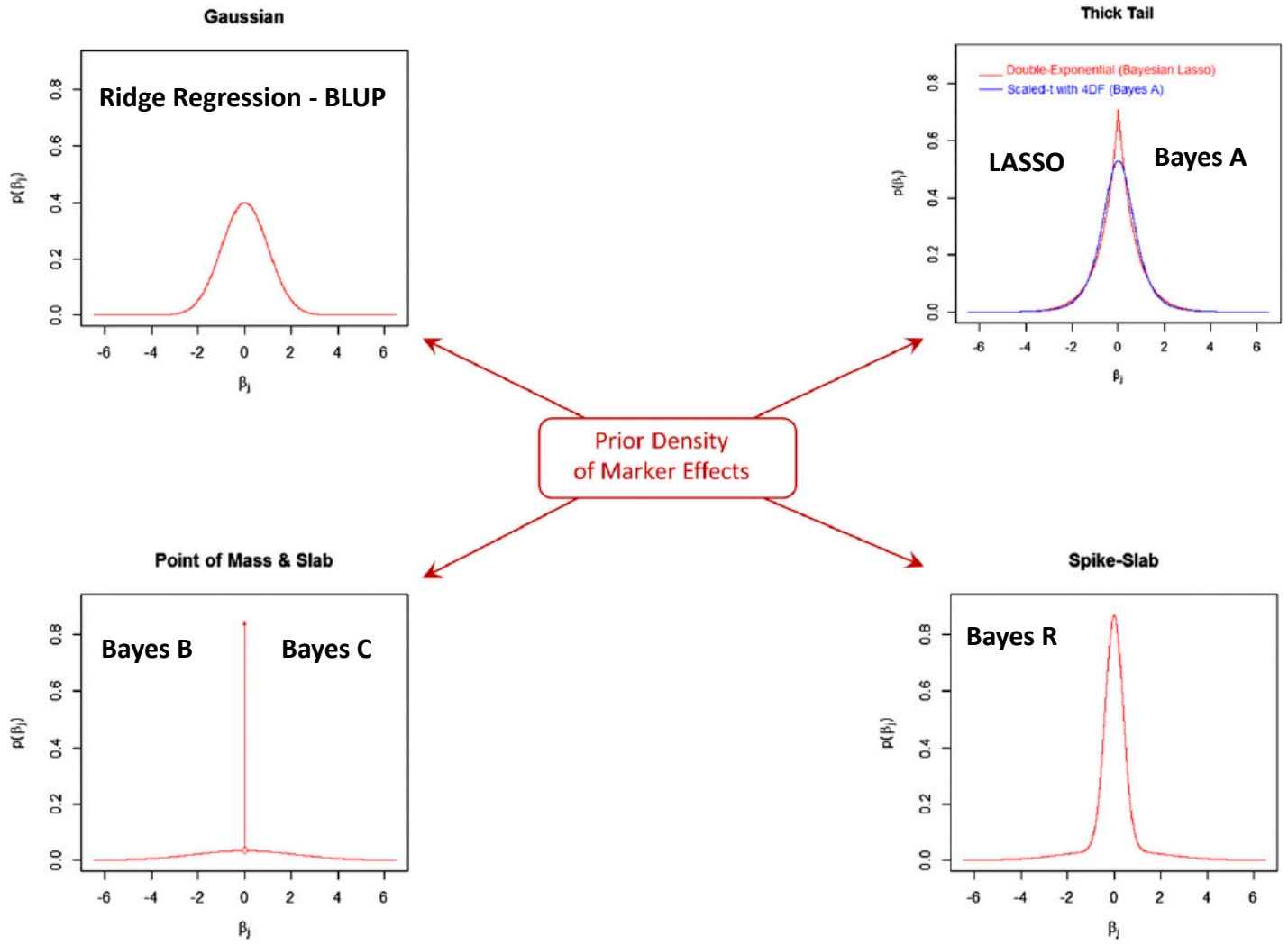
LASSO minimizes the function: $(\mathrm{y} - \mathbf{X}\mathrm{b})'(\mathrm{y} - \mathbf{X}\mathrm{b}) + \lambda\|\mathrm{b}\|$

- LASSO optimizes the absolute value of elements of b with the variation explained.
- Only elements of b that explain a large proportion of variance are allow to have a large absolute value

Penalized Methods

There are many models for dealing with over-parameterized data using penalized methods.

These methods vary in the underlying assumptions of the data distribution, which determines the types of penalties applied.



Gustavo de los Campos *et al*. 2013. "Whole-Genome Regression and Prediction Methods Applied to Plant and Animal Breeding." GENETICS *vol. 193 no. 2*

Dimension Reduction – Feature Selection

- Filter methods
  - Markers scored and ranked
- Wrapper methods
  - Often deploy optimized searching algorithms (Genetic Algorithm, Ant Colony Algorithm, Simulated Annealing)
  - Initialized based on some prior information
  - Select features, preform cross validation, and update based on predictive performance
- Work best when there are a small number of markers that explain a large proportion of variation.
  - Commonly used in disease diagnostics
  - Not as effective for complex, additive traits traits often modeling in plant and animal breeding.

Wrapper Approach using Swarm Intelligence

Real Ant Colony		Artificial Ant colony
m **ants searching for best rout to food source**	⟷	***m*** **ants searching for best subset of genes**
Communicate through a chemical pheromone trail	⟷	**Communicate through a PDF:**

$$P_{mc}(t) = \frac{(\tau_{mc}(t))^{\alpha} \eta_{mc}{}^{\beta}}{\sum_{m=1}^{nf} (\tau_{mc}(t))^{\alpha} \eta_{mc}{}^{\beta}}$$

Pheromone level changes each trip based on the time it takes to reach food source	⟷	**Pheromone level changes each iteration based on the prediction accuracy of selected genes**

$$\tau_m(t+1) = (1-\rho) * \tau_m(t) + \Delta\tau_m(t)$$

Dimension Reduction - Principle Component Regression.

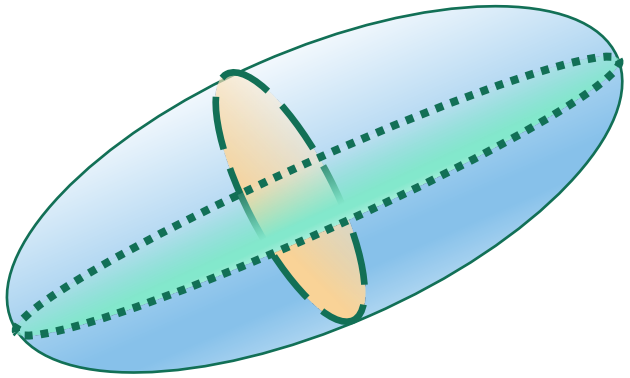
- The goal of PCR is to decompose **X** in *m* orthogonal vectors (we will call this matrix **T**). If there are correlations in the original vectors (columns of **X**) we can often find a small subset of vectors in **T** that explain most of the variance in **X**.

- We then use this reduced full rank matrix **T** to calculate OLS

- We can do this decomposition on **X** or **X'X** – here will will focus on the spectral decomposition (S.D.) of **X'X**

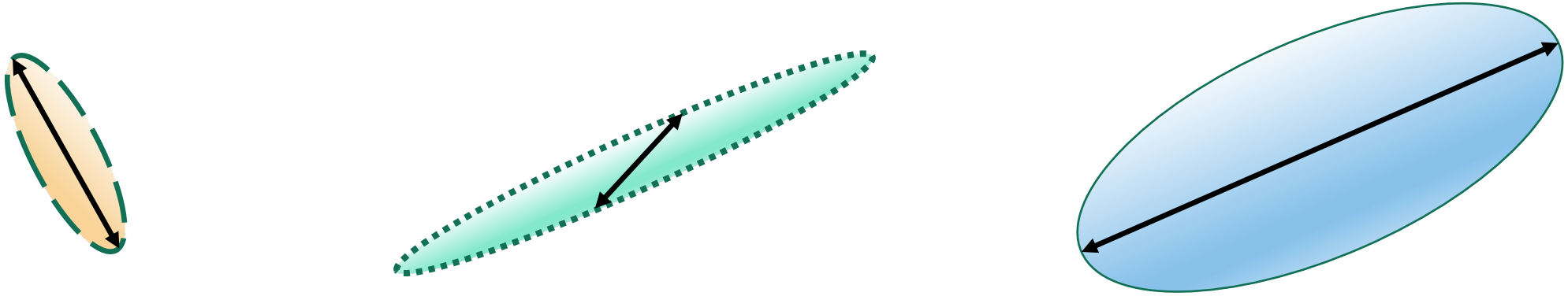
Principle Component Regression.

Orthogonalization – A process by which you take correlated vectors (columns in a matrix) and decompose them into orthogonal components.



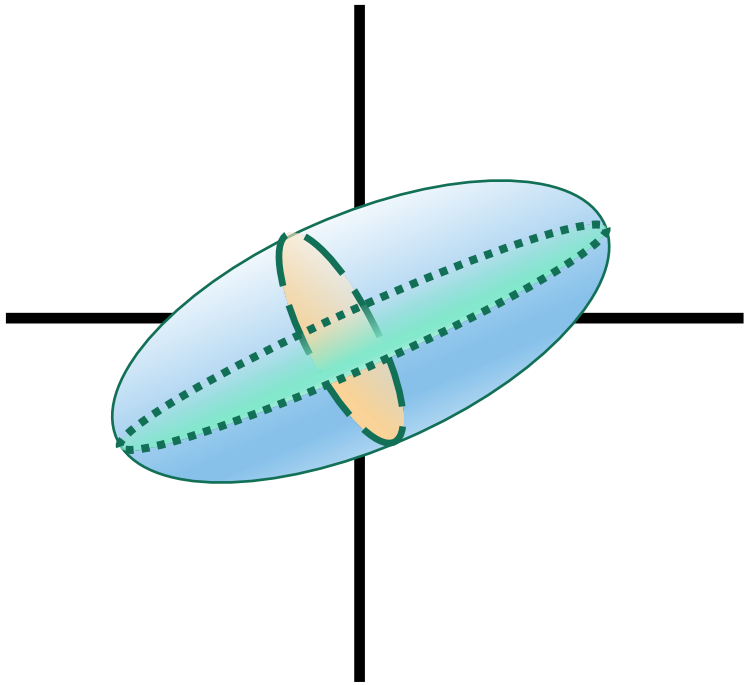
Principle Component Regression.

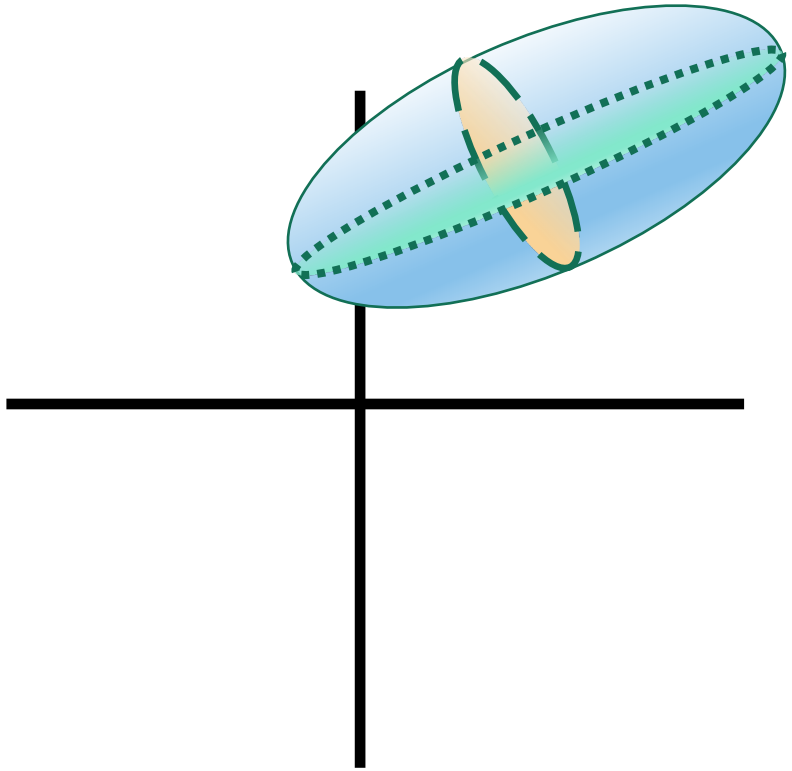
There are many ways to do orthogonalization – PCR uses an eigenvalue decomposition or spectral decomposition.



Principle Component Regression.

The first step in PCR is to center the matrix.





Spectral Decomposition

$\mathbf{X}$ is a *n x m* incidence matrix – *n* observations and *m* covariates

$\mathbf{X}'\mathbf{X}$ is a *m x m* $\quad S.D.(\mathbf{X}'\mathbf{X}) = \mathbf{UDU}' = \mathbf{X}'\mathbf{X}$

$\mathbf{U}$ is a symetric *m* x *m* matrix of eigen vectors $\quad \mathbf{UU}' = \mathbf{U}'\mathbf{U} = \mathbf{I}$

$\mathbf{D}$ is a *m* x *m* diagonal matrix of eigen values

Principle Component Regression.

To get **T** we need to multiple the matrix **X** by some matrix **P** to generate a matrix with orthogonal columns such that:

$\mathbf{T} = \mathbf{XP}$ and

$\mathbf{T}'\mathbf{T}$ is a diagonal matrix with non-zero values on the diagonal

Some Matrix Algebra …

$\mathbf{T'T = P'X'XP}$ Given -> $\mathbf{UDU' = X'X}$

$\mathbf{T'T = P'UDU'P}$ By setting **P** = **U**

$$\mathbf{T'T = P'UDU'P = U'UDU'U = D}$$

Using T for OLS

OLS using **X**

$$\hat{\mathbf{b}} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{y}$$

$$\hat{y} = \mathbf{X}\hat{\mathbf{b}}$$

T=XU

OLS using **T**

$$\hat{\mathbf{b}}^* = (\mathbf{T}'\mathbf{T})^{-1}\mathbf{T}'\mathbf{y} \longrightarrow \widehat{\mathbf{b}^*} = (\mathbf{D})^{-1}\mathbf{T}'\mathbf{y}$$

$$\hat{y} = \mathbf{T}\widehat{\mathbf{b}^*}$$

R Exercise - PCR

Dimension Reduction – Partial Least Squares

- In PCR we orthogonalize **X** to get a matrix **T** which contain most of the variation in **X** but is full rank.
- This approach is superior to feature selection when there are several highly correlated covariates all explaining some variation in **y**
- The draw back is that **y** is never considered when decomposing **X**
- Ideally, we would decompose **X** into orthogonal components that explain the most variation in **y** not **X.**
- **PLS differs from PCR in that we account for y when decomposing X.**

Nonlinear Iterative Partial Least Squares

- NIPALS is an iterative algorithm to decompose a matrix **X**
- Going back to the PCR approach we could decompose X as follows:

Initialize $\mathrm{t} \coloneqq \mathrm{x}_j$ for some column *j* in **X**

Loop until t converges

$$\mathrm{p} \coloneqq \frac{\mathbf{X}'\mathrm{t}}{\|\mathbf{X}'\mathrm{t}\|}$$

$$\mathrm{t} \coloneqq \mathbf{X}\mathrm{p}$$

Then set $\mathbf{X} \coloneqq \mathbf{X} - \mathrm{tp}'$

Repeat until you have all columns in $\mathbf{T}$

Partial Least Squares

- For PLS we substitute in **y** to give the following algorithm:

Loop until t converges

$$\mathrm{p} \coloneqq \frac{\mathbf{X}'\mathrm{y}}{\|\mathbf{X}'\mathrm{y}\|}$$

$$\mathrm{t} \coloneqq \mathbf{X}\mathrm{p}$$

Then set $\mathbf{X} \coloneqq \mathbf{X} - \mathrm{tp}'$

Repeat until you have all columns in $\mathbf{T}$

The above algorithm is for a single response variable – this can be generalized to multiple response variables – see notes on Canvas

Questions

Course Review