Regularization and Dimension Reduction for the Gene Expression Data
Gene Expression Data

What is our goal with the Gene Expression data?

1. Identify which genes are associated with higher/lower scale of malignant tumors.

Why is this goal important?

• “Turn off” genes associated with malignant tumors to prevent them from forming.
Gene Expression Data

How are we going to use statistics to achieve this goal?

1. Linear regression model.

What issues do we have to deal with in using a linear model for the GDP data?

1. \( P > n \) \( \Rightarrow \) we only have 102 patients but we have 5150 possible predictors so \( X'X \) is singular.
2. Potential collinearity.
3. Maybe there are outliers.
4. Possibly non-linear relationships?

Can we really assess #2-#4 with so few data points?
Probably not due to the curse of dimensionality.
Consider the following example: Assume $n$ points are distributed uniformly in the unit cube in $\mathbb{R}^P$. What is the side-length $\ell$ of a sub-cube that is expected to contains the proportion $\rho$ of the data? Answer: $\ell = \sqrt[\rho]{P}$

Black: $P = 1$
Green: $P = 2$
Red: $P = 8$
Blue: $P = 16$
Curse of Dimensionality

Curse of Dimensionality (Intuition): As dimension grows, data becomes sparse (scattered).

What can go wrong in high dimensions?
Overfitting - in high dimensions, we don’t have much local information and overreact to small bumps in the data resulting in poor test MSE.
“False Positives” – we can associate variables that aren’t related.

How do we get around this?
Need to devise a low-dimensional representation (e.g. regression with few coefficients). (Sparsity or parsimony)
Curse of Dimensionality

In terms of Gene Expression Data:

\[
\begin{align*}
\mathbf{y} &\sim_{n \times 1} \mathbf{X} \beta \sim_{n \times (P+1)} (P+1)_{1 \times 1} \epsilon \sim_{n \times 1}
\end{align*}
\]

which, when \( P > n \), yields an infinite number of solutions to the estimating equations:

\[
(X'X) \beta = X'y
\]

We need to add constraints to come up with a solution!

Constraints can be imposed via:

1. Variable selection.
2. Shrinkage/Regularization (penalized least squares).
3. Combining all predictors into \( M << P \) variables.
Shrinkage/Regularization

General Idea: Minimize not only squared residuals but coefficient size.

Let $\hat{\beta}$ minimize:

$$\min_{\beta} \sum_{i=1}^{n} (y_i - x'_i \beta)^2 + \lambda \sum_{p=1}^{P} \text{Size}(\beta_p)$$

or, equivalently,

$$\min_{\beta} \sum_{i=1}^{n} (y_i - x'_i \beta)^2 \quad \text{subject to} \quad \sum_{p=1}^{P} \text{Size}(\beta_p) = s(\lambda)$$
Shrinkage/Regularization

Two common choices:

1. Ridge Regression: Size($\beta_p$) = $\beta_p^2$ which gives the closed form solution:
   $$\hat{\beta}_{\text{ridge}} = (X'X + \lambda I)^{-1} X'(y - \bar{y})$$
   where $X$ does NOT have a column of 1s.

2. LASSO: Size($\beta_p$) = $|\beta_p|$

3. Elastic Net:
   $$\text{Size}(\beta_p) = \alpha|\beta_p| + (1 - \alpha)\beta_p^2, \alpha \in [0, 1]$$

Just FYI:  $$\sum_{p=1}^{P} \beta_p^2 = \mathcal{L}^2\text{-norm}, \quad \sum_{p=1}^{P} |\beta_p| = \mathcal{L}^1\text{-norm}$$
Shrinkage/Regularization

LASSO  Ridge
Shrinkage/Regularization

Ridge

LASSO
Shrinkage/Regularization

Why does this help us with the curse of dimensionality?

1. Restricts the flexibility of our model.
2. Including lots of covariates only increases the shrinkage penalty.

Why is penalized least squares preferred to least squares?

\[ \mathbb{E}_y \left[ (y - \hat{f}(x))^2 \right] = \text{Var} \left( \hat{f}(x) \right) + \left[ \text{Bias} \left( \hat{f}(x) \right) \right]^2 + \text{Var}(\epsilon) \]

Penalized least squares sacrifices some bias to gain smaller variance.
Shrinkage/Regularization

Purple = MSE, Green = Variance, Black = Bias

Ridge

LASSO
Shrinkage/Regularization

An Interesting Property of the LASSO:

Constraint contours are pointed so intersection usually occurs at an axis

Result: LASSO will zero out coefficients. Ridge regression, generally, will not
Shrinkage/Regularization

Issues to Consider:

1. Scale Matters!
   - Shrinking is NOT in terms of “importance” but rather size.
   - Center and scale the x’s

\[ \tilde{x}_{ij} = \frac{x_{ij} - \bar{x}_j}{\sqrt{\frac{1}{n} \sum_{i=1}^{n} (x_{ij} - \bar{x}_j)^2}} \]

   - Note: you may want to “backtransform” to original scale (R code does this for you internally).
Shrinkage/Regularization

Issues to Consider:

2. Don’t “shrink” the intercept.
   • Centering and scaling x’s help with this too (intercept won’t change).
   • Center the y’s too.

3. How do we pick $\lambda$ and (maybe) $\alpha$?
   • Cross validation
Shrinkage/Regularization

Issues to Consider:

4. How do we do confidence intervals using ridge or LASSO?
   • Note that LASSO, as written, is a tool NOT a “model” (no distributional assumptions just linearity assumption) so we can’t derive sampling distributions.
   • This might be OK if all you are doing is prediction and feature selection but NOT OK if you want to do inference.
Shrinkage/Regularization

Issues to Consider:

4. How do we do confidence intervals using ridge or LASSO?

\[ \hat{\beta}_j \pm t^*_{n-k} \text{SE}(\hat{\beta}_j) \]

\[ k \equiv \text{Effective number of parameters} \]

For Ridge:

\[
\text{SE}(\hat{\beta}_p) = \sqrt{\text{Var} \left( \hat{\beta}_{\text{ridge}} \right)}_{pp} = \sqrt{\text{Var} \left( (X'X + \lambda I)^{-1} X'Y \right)}_{pp}
\]

\[ = \sqrt{\sigma^2 (X'X + \lambda I)^{-1} X'X (X'X + \lambda I)^{-1}}_{pp} \]

\[ k = \text{trace} \left( X (X'X + \lambda I)^{-1} X' \right) \]
Shrinkage/Regularization

Issues to Consider:

4. How do we do confidence intervals using ridge or LASSO?
   \[ \hat{\beta}_j \pm t^*_{n-k} \text{SE}(\hat{\beta}_j) \]
   \[ k \equiv \text{Effective number of parameters} \]

For LASSO: We don’t have closed form solution to rely on so how can we get SE?
   \[ \text{SE}(\hat{\beta}_p) \approx \text{Bootstrap}(\lambda) \]
   \[ k \approx \# \text{ of non-zero coefficients (including intercept)} \]
The Bootstrap

Bootstrap Algorithm (Repeat MANY times):
For \( b = 1, \ldots, B \) where \( B \) is large
1. Take a “bootstrap sample” of size \( K \) from the original \( n \) data points with replacement.
2. Calculate and retain \( \hat{\beta}^b \) (or any other quantity you’re interested in, e.g., a prediction).

Theory tells us

\[
\sqrt{\frac{1}{B - 1} \sum_{b=1}^{B} \left( \hat{\beta}^b_j - \bar{\hat{\beta}}_j \right)^2} \approx \text{SE}(\hat{\beta}_j)
\]
The Bootstrap

Bootstrapped Confidence Intervals:

Quantile Interval:

\( (\hat{\beta}_{0.025}^{\text{boot}}, \hat{\beta}_{0.975}^{\text{boot}}) \)

Centered Interval: If \( \hat{\beta} - \beta \approx \hat{\beta}_{\text{boot}} - \hat{\beta} \) then

\[
\Pr(\hat{\beta}_{0.025}^{\text{boot}} < \hat{\beta}_{\text{boot}} < \hat{\beta}_{0.975}^{\text{boot}}) = \Pr(\hat{\beta}_{0.025}^{\text{boot}} - \hat{\beta} < \hat{\beta}_{\text{boot}} - \hat{\beta} < \hat{\beta}_{0.975}^{\text{boot}} - \hat{\beta}) \\
\approx \Pr(\hat{\beta}_{0.025}^{\text{boot}} - \hat{\beta} < \hat{\beta} - \beta < \hat{\beta}_{0.975}^{\text{boot}} - \hat{\beta}) \\
= \Pr(2\hat{\beta} - \hat{\beta}_{0.975}^{\text{boot}} < \beta < 2\hat{\beta} - \hat{\beta}_{0.025}^{\text{boot}}) \\
= 0.95
\]
The Bootstrap
Shrinkage/Regularization

Which do we use? LASSO, Ridge or Elastic Net?

LASSO:
1. Does variable selection.
2. Outperforms Ridge when coefficients are mostly zero.
3. Chooses one of correlated variables (no preference)

Ridge:
1. No variable selection.
2. Outperforms LASSO when lots of small coefficients.
3. Shrinks correlated X’s towards one another.

Elastic Net: Balances the two
Shrinkage/Regularization
Assumptions

What assumptions does Ridge/LASSO/Elastic use?

\[
\min_{\beta} \sum_{i=1}^{n} (y_i - x'_i \beta)^2 + \lambda \sum_{p=1}^{P} \text{Size}(\beta_p)
\]

1. Linear – yes
2. Independent – sort of: looks at one observation at a time
3. Normal – no
4. Equal variance – no
Shrinkage/Regularization

Bayesian Ridge Regression:

\[ y \sim \mathcal{N} (\beta_0 1 + X\beta, \sigma^2 I_n) \]
\[ \beta \sim \mathcal{N} (0, \lambda^{-1} \sigma^2 I_n) \]

\[
\log ([\beta \mid -]) \propto -\frac{1}{2\sigma^2} \left( \sum_{i=1}^{n} (y_i - x_i'\beta)^2 - \lambda \sum_{p=1}^{P} \beta_p^2 \right)
\]

Ridge Solution = Maximum *a posteriori* (MAP; mode) estimate of the Bayesian posterior distribution.
Shrinkage/Regularization

Bayesian LASSO:
\[ y \sim N(\beta_01 + X\beta, \sigma^2 I_n) \]
\[ \beta_p \stackrel{iid}{\sim} \text{Double Exponential or Laplace}(0, \lambda^{-1}) \]
\[ [\beta_j] \propto \exp\left\{ -\lambda |\beta_j|/\sigma^2 \right\} \]

\[
\log (|\beta| - ) \propto -\frac{1}{2\sigma^2} \left( \sum_{i=1}^{n} (y_i - x_i'\beta)^2 - \lambda \sum_{p=1}^{P} |\beta_p| \right)
\]

LASSO Solution = Maximum \textit{a posteriori} (MAP; mode) estimate of the Bayesian posterior distribution. Note, full Bayesian LASSO won’t do variable selection.
Dimension Reduction

General Idea: Combine $P$ into $M \ll P$ such that the information in the $M$ is roughly the same as the information in the original $P$ variables.

Define:

$$X_i = (X_{i1}, \ldots, X_{iP})'$$
$$Z_i = (Z_{i1}, \ldots, Z_{iM})'$$
$$Z_{im} = \psi_{m1}X_{i1} + \cdots + \psi_{mP}X_{iP}$$
$$= \psi'_m X_i$$

$$Z_i = \Psi'X_i$$

$$\Psi = \begin{pmatrix} \psi_{11} & \cdots & \psi_{1M} \\ \vdots & \ddots & \vdots \\ \psi_{P1} & \cdots & \psi_{PM} \end{pmatrix}_{P \times M}$$

$$y_i \sim \mathcal{N}(\theta_0 + Z'_i \theta, \sigma^2)$$
\( \theta_1, \ldots, \theta_M \) are not \( \beta_1, \ldots, \beta_P \)!

Recall \( Z_i' = X_i' \Psi \), then

\[
y_i = \theta_0 + Z_i' \theta + \epsilon_i \\
= \underbrace{\theta_0 + X_i' \Psi \theta}_{\beta_0} + \epsilon_i
\]

So \( \beta = (\theta_0, \Psi \theta)' \)
Dimension Reduction

Why does this work?

It enforces the constraint:

\[ \beta = (\theta_0, \Psi \theta)' \]

Which increases bias but reduces variance.
Dimension Reduction

What do we want $\Psi$ to be?

Principal Component Regression: We’d like (or need):

1. If $Z_i = \Psi'X_i$ then we’d like $X_i \approx (\Psi')^{-1}Z_i \ \forall i$ so that the information we created in $Z$ pretty much captures everything important in $X$ (which also means that the inverse should exist).

2. For computational reasons, it would be nice if:

$$\text{Var}(Z_i) = \text{diag}(\sigma^2_{Z_1}, \ldots, \sigma^2_{Z_M}) = D_z$$

so we don’t have to worry about collinearity.

Thinking about these together, choose $\Psi$ such that

$$\text{Var}(X_i) = \Sigma \approx (\Psi')^{-1}D_z((\Psi')^{-1})' = CD_zC'$$

$$C = (\Psi')^{-1}$$
Dimension Reduction

Eigenvalue (Spectral) Decomposition of a Matrix:
Any positive definite matrix $\Sigma_x$ can be written as

$$\Sigma_x = CDC'$$

where $C$ is orthonormal (i.e. $C'C = I$) matrix of eigenvectors as columns and $D$ is diagonal matrix of eigenvalues.
Dimension Reduction

**Principal Component Solution:**
Set \( \mathbf{C} \) to be the first \( M \) columns of the matrix of eigenvectors of \( \Sigma_x \). Then, \( \Psi = (\mathbf{C}^{-1})' = \mathbf{C} \)

\[
\Psi = \text{eigen(cov} (\mathbf{X})) \text{vectors}[1 : M]
\]

\[
\begin{bmatrix}
\mathbf{Z} \\
\mathbf{X}
\end{bmatrix}_{N \times M} =
\begin{bmatrix}
\mathbf{X} \\
\mathbf{Y}
\end{bmatrix}_{N \times P} \Psi
\]

\[
\mathbf{y} = \beta_0 + \mathbf{Z} \theta + \epsilon
\]

\[
\beta = \Psi \theta
\]

\[
\hat{\beta} = \Psi \hat{\theta}
\]
Dimension Reduction

Properties of Principal Component Solution:
1. First principal component is direction along which is the highest variation. Also, line closest to data.
Dimension Reduction

Properties of Principal Component Solution:

2. Second principal component is next direction, orthogonal to first, along which has the next highest variation. “Next closest orthogonal line”

Definitions:

1. $Z_{im}$: the principal component score for the $i^{th}$ observation.
2. $\psi_{pm}$: $m^{th}$ loading for the $p^{th}$ variable.
Dimension Reduction

Principle Component Regression in Practice:

1. We don’t know $\Sigma_x$ so we replace it with

$$\hat{\Sigma}_x = \frac{1}{n - 1} \sum_{i=1}^{n} (\mathbf{x}_i - \bar{\mathbf{x}})(\mathbf{x}_i - \bar{\mathbf{x}})'$$

2. Scale matters so center and scale variables first

$$\tilde{x}_{ij} = \frac{x_{ij} - \bar{x}_j}{\sqrt{\frac{1}{n} \sum_{i=1}^{n} (x_{ij} - \bar{x}_j)^2}}$$
Dimension Reduction

Principle Component Regression in Practice:

3. How do we choose the number of variables $M$?
   - Forward variable selection
   - Cross validation
   - Note that:
     \[ \sum_{m=1}^{M} \sigma_{Z_m}^2 = \text{trace}(D_z) \approx \text{trace}(\Psi' \Sigma_x \Psi) \]
     \[ = \text{trace}(\Sigma_x) \]
     choose $M$ so that
     \[ \frac{\sum_{m=1}^{M} \sigma_{Z_m}^2}{\sum_{m=1}^{P} \sigma_{Z_m}^2} \approx \frac{\sum_{m=1}^{M} \sigma_{Z_m}^2}{\sum_{p=1}^{P} \text{Var}(X_p)} \approx 100\% \]
Dimension Reduction

Principle Component Regression in Practice:

4. Recall our model is

\[ y_i \overset{iid}{\sim} \mathcal{N}(\theta_0 + Z_i' \theta, \sigma^2) \]

so be sure to “back-transform” from \( \hat{\theta} \) to \( \hat{\beta} \) via

\[ \hat{\beta} = \Psi \hat{\theta} \]

and

\[ \text{Var}(\hat{\beta}) = \Psi \text{Var}(\hat{\theta}) \Psi' = \sigma^2 \Psi (Z'Z)^{-1} \Psi' \]

CI : \( \hat{\beta}_p \pm t^*_{n-M-1} \text{SE}(\hat{\beta}_p) \)
Dimension Reduction

Partial Least Squares:
• Note that in principal components, how we choose $\{\psi_{pm}\}$ does not depend on $y$.

General idea: set $\{\psi_{mp}\}$ to be estimate of slope in regression model.
Dimension Reduction

Partial Least Squares Algorithm:

Steps:

1. Let $X_p^{(1)} = X_p = (X_{1p}, \ldots, X_{np})'$.
2. Set $\{\psi_{1p}\} = \{\hat{\beta}_p\}$ in model $y_i = \beta_0 + \beta_p X_{ip} + \epsilon_i$ and calculate:
   $$z_{i1} = \sum_{p=1}^{P} \psi_{1p} X_{ip}$$

3. For m=2,…,M
   a) Orthogonalize covariates with respect to $Z_{m-1}$:
      $$X_p^{(m)} = X_p^{(m-1)} - Z_{(m-1)} (Z'_{(m-1)} Z_{(m-1)})^{-1} Z'_{(m-1)} X_p^{(m-1)}$$
   b) Set $\{\psi_{mp}\} = \{\hat{\beta}_p\}$ in model $y_i = \beta_0 + \beta_p X_{ip}^{(m)} + \epsilon_i$ and calculate $z_{im} = \sum_{p=1}^{P} \psi_{mp} X_{ip}$. 
Partial Least Squares (solid) vs. Prin. Components (dashed):
Partial Least Squares in Practice:

1. Recall our model is

$$y_i \overset{iid}{\sim} \mathcal{N}(\theta_0 + Z_i'\theta, \sigma^2)$$

so be sure to “back-transform.”

Note: You can always bootstrap to get standard errors.
Dimension Reduction

Partial Least Squares in Practice:

2. We still need to standardize quantitative variables via

\[ \tilde{x}_{ij} = \frac{x_{ij} - \bar{x}_j}{\sqrt{\frac{1}{n} \sum_{i=1}^{n} (x_{ij} - \bar{x}_j)^2}} \]

3. Can have smaller bias but variance is generally higher than using principal components.

4. How do we choose \( M \)?
   - Forward variable selection
   - Cross validation
Dimension Reduction

Partial Least Squares Vs. Principle Components:

Principle Components:
  1. Orthogonal Components
  2. Stable – low variance
  3. Captures variability in X

Partial Least Squares
  1. Orthogonal Components
  2. High(er) Variance due to collinearity
  3. Captures variability in Y given X
Final Thoughts:

1. If you have two highly collinear variables you can combine them (reduce their dimension) via principal components to get rid of the collinearity.
2. You can use these methods anytime you do regression (P doesn’t necessarily have to be greater than N – that just happened to be the case here).
Expectations for Gene Expression Analysis

1. Description of linear model and issues we have in using it (curse of dimensionality).

2. Without using variable selection – so you have to use a regularization or dimension reduction technique. Describe the approach you use.

3. Justify choices in regularization (e.g. penalty parameter) or number of reduced dimensions (e.g. M).

4. Don’t look at interactions – you don’t have enough data.

5. There are 4 different approaches – all of which are reasonable. Compare the results from at least 2.