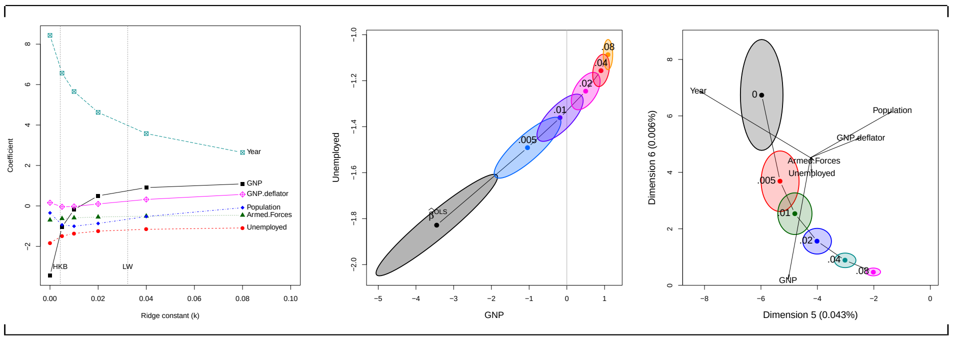
Generalized Ridge Trace Plots

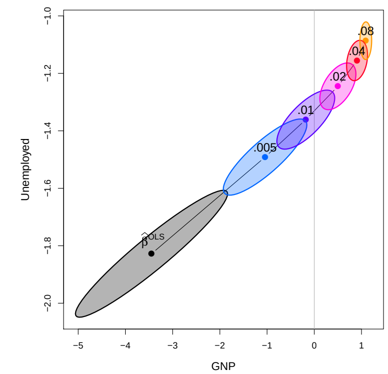
Visualizing Bias *and* Precision with the genridge R package

Michael Friendly

SCS Seminar
Jan, 2011



Outline



Bivariate ridge trace plot for GNP & Unemployed in Longley data

Ridge Regression and Shrinkage Methods: Bias vs. Precision

- Consider the univariate classical linear model,

$$\mathbf{y} = \beta_0 \mathbf{1} + \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon},$$

  where $E(\boldsymbol{\epsilon}) = \mathbf{0}$, $\text{Var}(\boldsymbol{\epsilon}) = E(\boldsymbol{\epsilon}\boldsymbol{\epsilon}^{\mathsf{T}}) = \sigma^2 \mathbf{I}$,
- Under moderate to severe collinearity— high $R^2(X_i \,|\, \text{other Xs})$—
  - Standard errors of $\boldsymbol{\beta}$ are inflated
  - OLS estimates of $\boldsymbol{\beta}$ tend to be too large on average
- Ridge regression and related shrinkage methods
  - Desire: increase precision (decrease $\widehat{\text{Var}}(\widehat{\boldsymbol{\beta}})$)
  - OLS estimates $\boldsymbol{\beta}$ are constrained, shrinking them toward $\boldsymbol{\beta}^{\mathsf{T}}\boldsymbol{\beta} = 0$
  - All methods use some tuning parameter (k) to quantify the tradeoff
  - How to choose? Numerical criteria, generalized cross-validation, bootstrap, etc.

Bias vs. Precision

- Particularly important when the goal is predictive accuracy
- In-sample prediction error typically descreases with increased model complexity
- For new samples, prediction error typically suffers from the high variance of complex models
  - But: how to visualize the tradeoff?

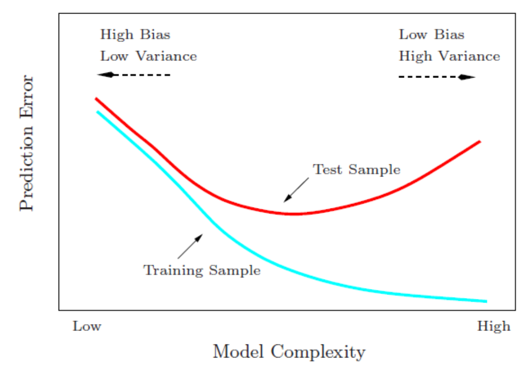
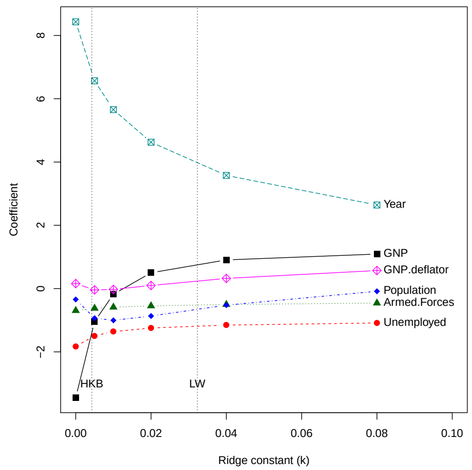


Fig. source: Hastie etal., *Elements of Statistical Learning*

Univariate ridge trace plots

- Typical: univariate line plot of β_k vs. shrinkage, k
- What can you see here regarding bias vs. precision?
- This is the wrong graphic form, for a multivariate problem!
- Goal: visualize $\widehat{\beta}_k$ vs. $\widehat{\text{Var}}(\widehat{\beta}_k)$



Caveats

- Ridge regression is not a panacea for problems of collinearity
  - Collinearity is a data problem — no magic cure.
  - Ridge regression is often more like palliative care — make the data as comfortable as possible with the disease.
  - Still, widely used in some contexts (small n, econometrics, chemistry & physics applications)
- Variable re-specification is often more effective
  - Normalize variables as ratios to adjust for GNP, population, etc.
  - Center variables in interactions and polynomial terms
  - Interpretable orthogonalization of related variables: as sums & differences, contrasts, Gram-Schmidt, ...
  - PCA regression cures the problem, but makes interpretation more difficult
- Thoughtful model re-specification is often helpful
- Nevertheless, the graphical ideas here are novel and extend to other model selection methods.

Motivating example: Longley data

Longley (1965) data: economic time series ($n = 16$) of yearly data from 1947 – 1962, often used as an example of extreme collinearity.

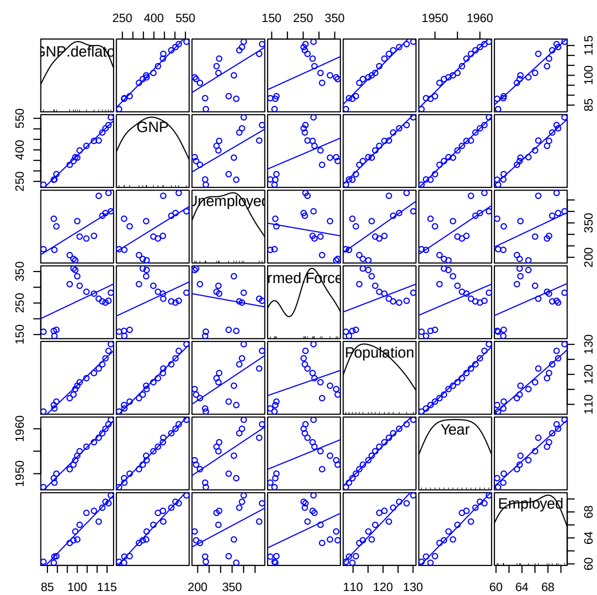
```
> names(longley)

[1] "GNP.deflator" "GNP"          "Unemployed"   "Armed.Forces"
[5] "Population"   "Year"         "Employed"
```

We take number of people Employed as the response:

```
> lmod <- lm(Employed ~ GNP + Unemployed + Armed.Forces +
      Population + Year + GNP.deflator, data = longley)
> vif(lmod)

      GNP  Unemployed Armed.Forces  Population     Year  GNP.deflator
 1788.513      33.619        3.589     399.151  758.981       135.532
```

As suspected, almost all VIFs are very large.


```
> library(car)
> scatterplotMatrix(longley, smooth=FALSE, col="blue", gap=0.2,
    cex.labels=1.2)
```



Historical sidebar on Longley's data

- Longley (1967) used used these data to demonstrate the effects of numerical instability and round-off error in least squares computations based on direct inversion of the crossproducts matrix, $(\mathbf{X}^\mathsf{T}\mathbf{X})^{-1}$.
- It sparked the development of more numerically stable algorithms (e.g., QR, modified Gram-Schmidt, etc.) now used is almost all statistical software.
- These data are perverse, in that there is clearly lack of independence and *structural* collinearity – GNP, Year, GNP.deflator, Population.
- Looking back, a scatterplot matrix would have revealed some of these problems...
- ... and perhaps made the example less compelling

Ridge Regression: Properties I

- OLS estimates:

$$\begin{aligned} \widehat{\beta}^{\text{OLS}} &= (\mathbf{X}^\mathsf{T}\mathbf{X})^{-1}\mathbf{X}^\mathsf{T}\mathbf{y} \ , \\ \widehat{\text{Var}}(\widehat{\beta}^{\text{OLS}}) &= \widehat{\sigma}^2(\mathbf{X}^\mathsf{T}\mathbf{X})^{-1}. \end{aligned}$$

- Ridge regression: replaces $\mathbf{X}^\mathsf{T}\mathbf{X}$ with $\mathbf{X}^\mathsf{T}\mathbf{X} + k\mathbf{I}$
  - drives $|\mathbf{X}^\mathsf{T}\mathbf{X} + k\mathbf{I}|$ away from zero even if $|\mathbf{X}^\mathsf{T}\mathbf{X}| \approx 0$.
  - drives $||\beta|| = (\beta^\mathsf{T}\beta)^{1/2}$ toward zero— increasing "bias"
  - decreases the "size" of $\widehat{\text{Var}}(\widehat{\beta})$— increasing precision— in that

$$|\widehat{\text{Var}}(\widehat{\beta}^{\text{OLS}})| \quad \geq \quad |\widehat{\text{Var}}(\widehat{\beta}_k^{\text{RR}})| \qquad \text{decreases with } k$$

Ridge Regression: Properties II

- Ridge estimates:

$$\begin{aligned} \widehat{\beta}_k^{\text{RR}} &= (\mathbf{X}^\mathsf{T}\mathbf{X} + k\mathbf{I})^{-1}\mathbf{X}^\mathsf{T}\mathbf{y} && (1) \\ &= \mathbf{G}_k\,\widehat{\beta}^{\text{OLS}} \ , \\ \widehat{\text{Var}}(\widehat{\beta}_k^{\text{RR}}) &= \widehat{\sigma}^2\mathbf{G}_k(\mathbf{X}^\mathsf{T}\mathbf{X})^{-1}\mathbf{G}_k^\mathsf{T} \ . && (2) \end{aligned}$$

  where $\mathbf{G}_k = \left[\mathbf{I} + k(\mathbf{X}^\mathsf{T}\mathbf{X})^{-1}\right]^{-1}$, the $(p \times p)$ "shrinkage" matrix.
- Equivalent to penalized LS criterion,

$$\text{RSS}(k) = (\mathbf{y} - \mathbf{X}\beta)^\mathsf{T}(\mathbf{y} - \mathbf{X}\beta) + k\beta^\mathsf{T}\beta \qquad (k \geq 0) \ , \qquad (3)$$

- Or, to a constrained LS minimization problem,

$$\widehat{\beta}^{\text{RR}} = \underset{\beta}{\text{argmin}}(\mathbf{y} - \mathbf{X}\beta)^\mathsf{T}(\mathbf{y} - \mathbf{X}\beta) \quad \text{subject to} \quad \beta^\mathsf{T}\beta \leq t(k) \qquad (4)$$

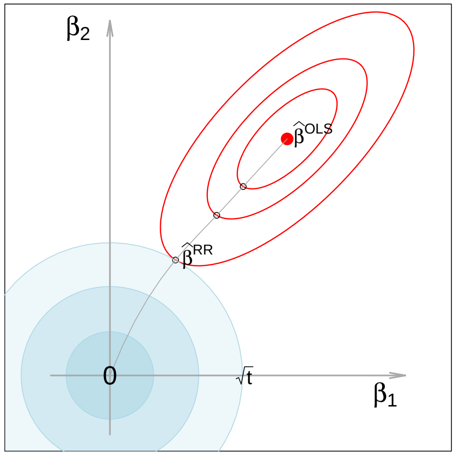
Ridge Regression: Geometry

Ridge regression solution has a simple geometric interpretation based on ellipsoids of the $RSS(k)$ function,

$$\text{RSS}(k) = (\mathbf{y} - \mathbf{X}\beta)^\mathsf{T}(\mathbf{y} - \mathbf{X}\beta) + k\beta^\mathsf{T}\beta$$

OLS coefficients are shrunk toward $\mathbf{0}$ along the locus of osculation of

- Covariance ellipsoid of β^{OLS}
- Unit sphere $\beta^\mathsf{T}\beta \leq t(k)$



The matrix $\mathbf{G}_k = \left[\mathbf{I} + k(\mathbf{X}^\mathsf{T}\mathbf{X})^{-1}\right]^{-1}$ shrinks the covariance matrix of β_k in a similar way

The genridge package: overview

Computation:

`ridge`	Calculates ridge regregression estimates; returns an object of class `"ridge"`
`pca.ridge`	Transform coefficients and covariance matrices to PCA/SVD space; returns an object of class `c("pcaridge", "ridge")`
`vif.ridge`	Calculates VIFs for `"ridge"` objects
`precision`	Calculates measures of precision and shrinkage

Plotting methods:

`traceplot`	Univariate ridge trace plots
`plot.ridge`	2D ridge trace plots
`pairs.ridge`	scatterplot matrix of ridge trace plots
`plot3d.ridge`	3D ridge trace plots
`biplot.ridge`	ridge trace plots in PCA/SVD space

The genridge package: ridge()

The function `ridge()` calculates ridge regression estimates

It also has a formula interface.

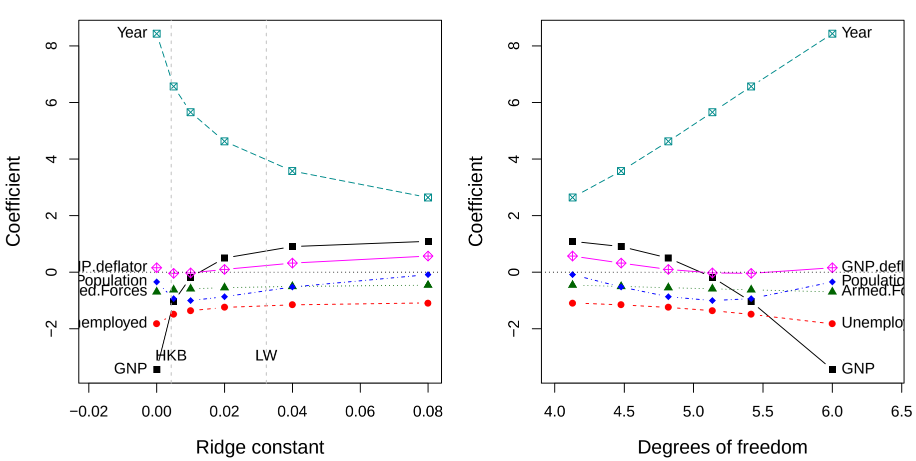
```
> library(genridge)
> longley.y <- longley[, "Employed"]
> longley.X <- model.matrix(lmod)[, -1]
> lambda <- c(0, 0.005, 0.01, 0.02, 0.04, 0.08)
> lridge <- ridge(longley.y, longley.X, lambda = lambda)
> coef(lridge)

          GNP Unemployed Armed.Forces Population  Year GNP.deflator
0.000 -3.4472     -1.828      -0.6962   -0.34420 8.432      0.15738
0.005 -1.0425     -1.491      -0.6235   -0.93558 6.567     -0.04175
0.010 -0.1798     -1.361      -0.5881   -1.00317 5.656     -0.02612
0.020  0.4995     -1.245      -0.5476   -0.86755 4.626      0.09766
0.040  0.9059     -1.155      -0.5039   -0.52347 3.577      0.32124
0.080  1.0907     -1.086      -0.4583   -0.08596 2.642      0.57025
```

It returns a `"ridge"` object containing coefficients, covariance matrices and other quantities:

```
> names(lridge)

 [1] "lambda" "df"     "coef"   "cov"    "mse"    "scales" "kHKB"
 [8] "kLW"    "svd.D"  "svd.U"  "svd.V"
```


Univariate ridge trace plots: traceplot()

```
> traceplot(lridge, cex.lab=1.25, xlim=c(-.01, 0.08))
> traceplot(lridge, X="df", cex.lab=1.25, xlim=c(4,6.2))
```



As will be explained, the ridge constant *k* can also be parameterized in terms of effective degrees of freedom.

Variance Inflation Factors: vif() method

`vif()` for a `"ridge"` object calculates variance inflation factors for all values of the ridge constant

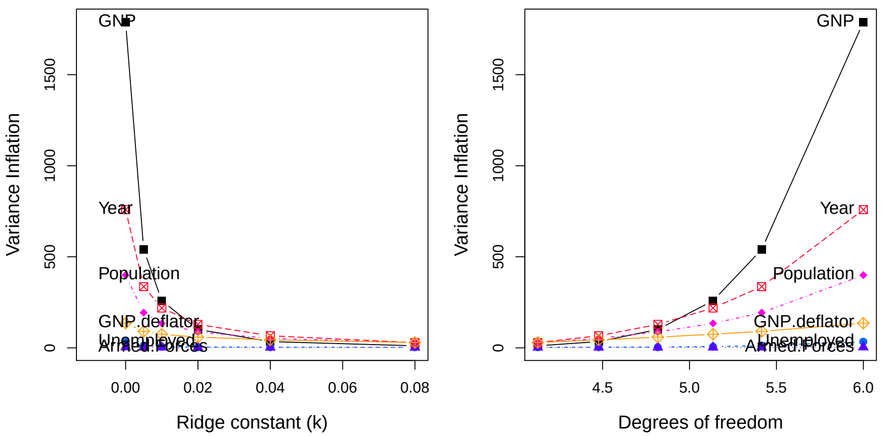
```
> vridge <- vif(lridge)
> vridge

          GNP Unemployed Armed.Forces Population   Year GNP.deflator
0.000 1788.51     33.619        3.589     399.15 758.98       135.53
0.005  540.04     12.118        2.921     193.30 336.15        90.63
0.010  259.00      7.284        2.733     134.42 218.84        74.79
0.020  101.12      4.573        2.578      87.29 128.82        58.94
0.040   34.43      3.422        2.441      52.22  66.31        43.56
0.080   11.28      2.994        2.301      28.59  28.82        29.52
```

This gives some idea of the effect of shrinkage on variance inflation

Variance Inflation Factors: Ridge VIF plots?

Plots of VIF vs k for individual variables show the magnitude of problems, but suffer from being swamped by the largest value.

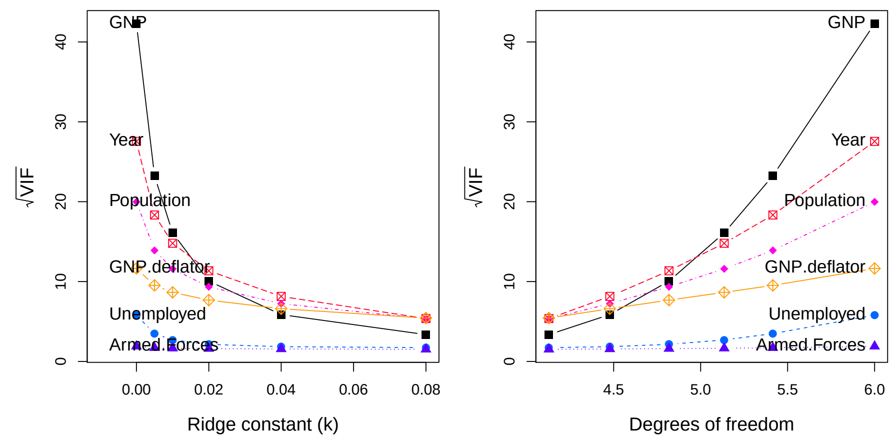
```
> matplot(rownames(vridge), vridge, type = "b", xlab = "Ridge constant (k)"
      ylab = "Variance Inflation", xlim = c(-0.01, 0.08),
      col = clr, pch = pch, cex = 1.2, cex.lab = 1.25)
```



Variance Inflation Factors: Ridge VIF plots?

At the very least, plot $\sqrt{VIF}$, which is the multiplier for standard errors

```
> matplot(rownames(vridge), sqrt(vridge), type = "b", xlab = "Ridge constan
      ylab = expression(sqrt(VIF)), xlim = c(-0.01, 0.08),
      col = clr, pch = pch, cex = 1.2, cex.lab = 1.25)
```



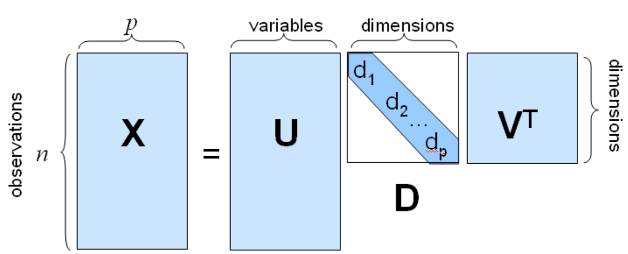
Ridge Regression: SVD I

Naive formula for $\widehat{\beta}_k^{\mathrm{RR}}$, Eqn. (1), is computationally expensive, numerically unstable and conceptually opaque

- Alternative formulation in terms of the SVD of $\mathbf{X}$:

$$\underset{(n\times p)}{\mathbf{X}} = \underset{(n\times p)}{\mathbf{U}} \; \underset{(p\times p)}{\mathbf{D}} \; \underset{(p\times p)}{\mathbf{V}}^{\mathsf{T}} \tag{5}$$

where $\mathbf{U}^{\mathsf{T}}\mathbf{U} = \mathbf{V}^{\mathsf{T}}\mathbf{V} = \mathbf{I}$, and $\mathbf{D} = \mathrm{diag}\,(d_1, d_2, \ldots d_p)$ is the diagonal matrix of ordered singular values.



Ridge Regression: SVD II

- The ridge estimates can then be calculated efficiently as

$$\widehat{\beta}_k^{\mathrm{RR}} = (\mathbf{D}^2 + k\mathbf{I})^{-1}\mathbf{D}\mathbf{U}^{\mathsf{T}}\mathbf{y} = \left(\frac{d_i}{d_i^2 + k}\right) \mathbf{u}_i^{\mathsf{T}}\mathbf{y}, \quad i = 1, \ldots p \tag{6}$$

- Fitted values can be expressed as

$$\widehat{\mathbf{y}}_k^{\mathrm{RR}} = \mathbf{X}(\mathbf{X}^{\mathsf{T}}\mathbf{X} + k\mathbf{I})^{-1}\mathbf{D}\mathbf{U}^{\mathsf{T}}\mathbf{y} = \sum_i^p \mathbf{u}_i \left(\frac{d_i^2}{d_i^2 + k}\right) \mathbf{u}_i^{\mathsf{T}}\mathbf{y}$$

- The factors $d_i^2/(d_i^2 + k) \leq 1$ indicate the degree of shrinkage wrt the orthonormal basis of the column space of $\mathbf{X}$ given by $\mathbf{U}$.
- The rows of $\mathbf{V}^{\mathsf{T}}$ give the linear combinations of the variables for each dimension— we use this for biplot views.

Ridge Regression: SVD III

- Small singular values d_i correspond to directions (rows of $\mathbf{V}^\mathsf{T}$) which ridge regression shrinks the most.
- These are the directions which contribute most to collinearity
- Gives an alternative characterization of the ridge tuning parameter (k) in terms of effective degrees of freedom

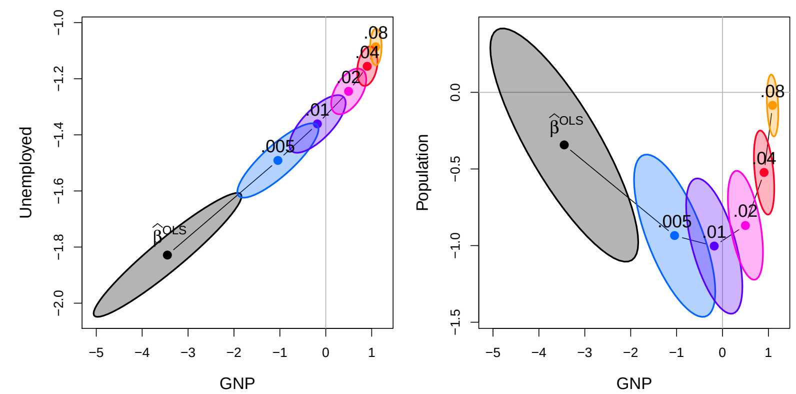
$$\mathrm{df}_k = \mathrm{tr}[\mathbf{X}(\mathbf{X}^\mathsf{T}\mathbf{X} + k\mathbf{I})^{-1}\mathbf{X}^\mathsf{T}] = \sum_i^p \left(\frac{d_i^2}{d_i^2 + k} \right) \le p \qquad (7)$$

- Eqn. (7) follows from the fact that, for OLS, the hat matrix $\mathbf{H} = \mathbf{X}(\mathbf{X}^\mathsf{T}\mathbf{X})^{-1}\mathbf{X}^\mathsf{T}$ has $\mathrm{tr}(\mathbf{H}) = p$, the number of parameters, dimensions, or df.

Generalized Ridge Trace Plots: Main idea

Rather than plotting just the univariate trajectories of β_k vs. k, plot the covariance ellipsoids of $\widehat{\boldsymbol{\Sigma}}_k \equiv \widehat{\mathrm{Var}}(\widehat{\boldsymbol{\beta}}_k)$ over same range of k

- Centers of the ellipsoids are $\widehat{\boldsymbol{\beta}}_k$ – same info as in univariate plot
- Can see how change in one coefficient is related to changes in others
- Relative size & shape of ellipsoids shows directly effect on precision



Generalized Ridge Trace Plots: Possible views

For a given data set, assume we have a set of K ellipsoids, $\mathcal{E}(\widehat{\boldsymbol{\beta}}_{k_j}, \widehat{\boldsymbol{\Sigma}}_{k_j})$, $j = 1, 2, \ldots, K$, each of dimension p.

These can be viewed in a variety of ways:

- Calculate summary measures of variance (size of $\boldsymbol{\Sigma}_k$) and shrinkage (size of $\boldsymbol{\beta}_k$) and plot directly or vs. k
- 2D views of the projections of the ellipsoids for pairs of predictors
- Scatterplot matrix for all pairwise 2D views
- 3D views of projections for triples of predictors
- Informative 2D/3D views projected into PCA/SVD space
- Interactive, dynamic graphics for any of the above allowing choice of shrinkage factors, etc. via software controls

Measuring Precision and Shrinkage: precision()

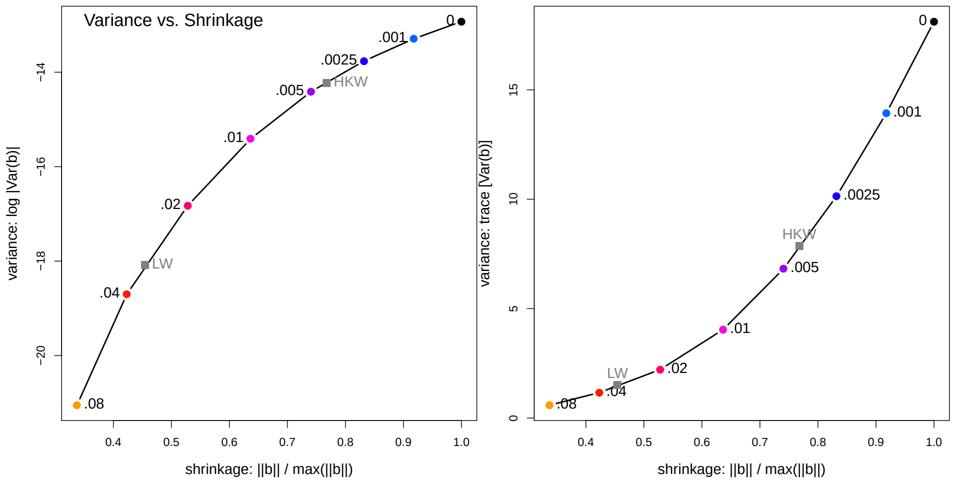
Other benefits of this multivariate approach:

- Shrinkage ("bias") can be measured by the length of the coefficient vector, $||\boldsymbol{\beta}|| = \sqrt{\boldsymbol{\beta}^\mathsf{T}\boldsymbol{\beta}}$
- Variance (inverse precision) can be measured by the "size" of the covariance ellipsoid, as functions of its eigenvalues, $\lambda_i, i = 1, \ldots, p$.
  - Linearized volume: $\log|\boldsymbol{\Sigma}_k|$ or $|\boldsymbol{\Sigma}_k|^{1/p} = \sqrt[p]{\prod \lambda_i}$ ~ Wilks Λ
  - Average measure of size: $\mathrm{tr}(\boldsymbol{\Sigma}_\mathbf{k}) = \sum \lambda_i$ ~ Pillai trace
  - Maximum dimension: λ_i ~ Roy's max root

```
> (pdat <- precision(lridge))

      lambda    df     det    trace max.eig norm.beta
0.000  0.000 6.000  -12.93  18.1190 15.4191     3.807
0.005  0.005 5.415  -14.41   6.8209  4.6065     2.819
0.010  0.010 5.135  -15.41   4.0423  2.1807     2.423
0.020  0.020 4.818  -16.83   2.2180  1.0255     2.011
0.040  0.040 4.478  -18.70   1.1647  0.5808     1.611
0.080  0.080 4.128  -21.05   0.5873  0.2599     1.284
```

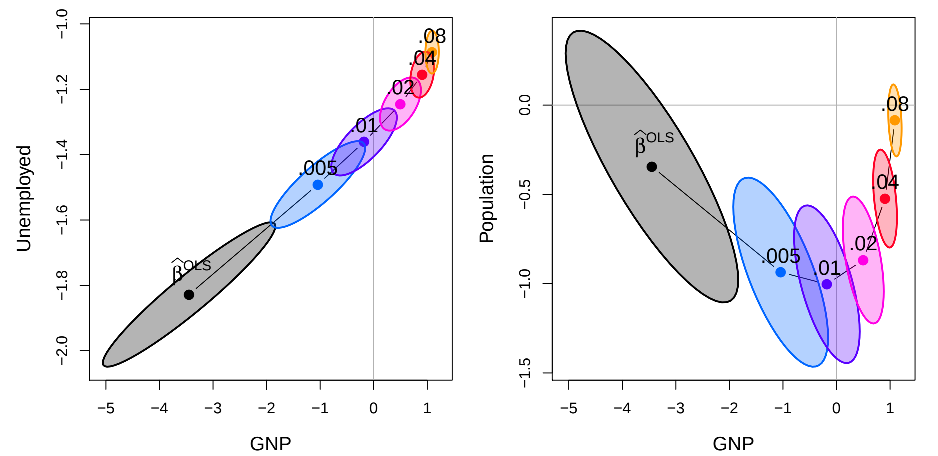
Visualizing Precision vs. Shrinkage

```
> with(pdat, {
      plot(norm.beta, det, type = "b", cex.lab = 1.25,
          pch = 16, cex = 1.5, col = clr, xlab = "shrinkage: ||b||",
          ylab = "variance: log |(Var(b)|)")
      text(norm.beta, det, lambdaf, cex = 1.25, pos = 2)})
```



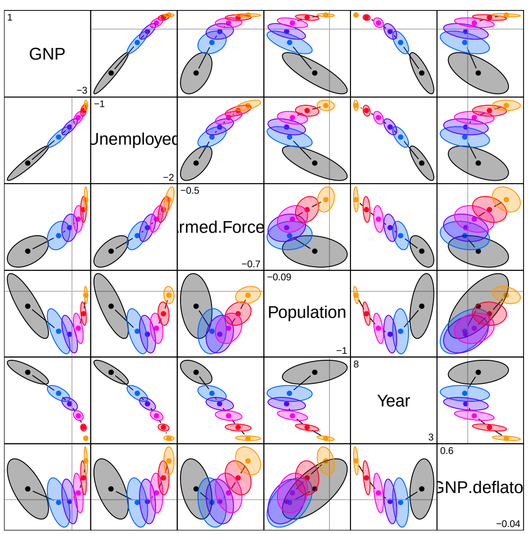
Bivariate ridge trace plots: plot() method

```
> clr <- c("black", rainbow(5, start=.6, end=.1))
> plot(lridge, var=c(1,2), radius=0.5, col=clr, cex.lab=1.25, fill=TRUE)
> plot(lridge, var=c(1,4), radius=0.5, col=clr, cex.lab=1.25, fill=TRUE)
```



Scatterplot matrix of ridge trace plots: pairs() method

```
> pairs(lridge, radius=0.5, diag.cex=1.75, col=clr, fill=TRUE)
```



Bivariate ridge trace plots: Observations

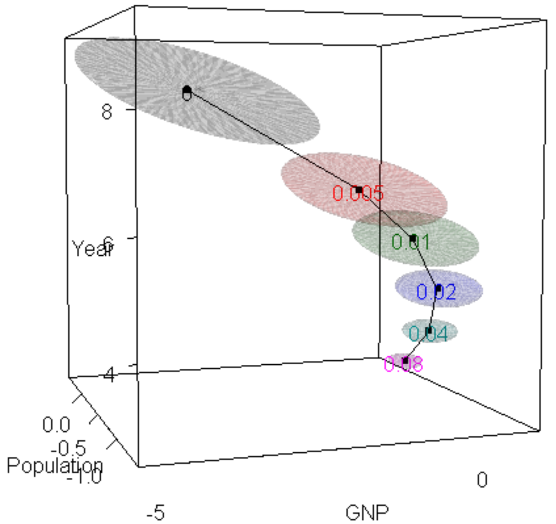
Bivariate ridge trace plots show a variety of things that cannot be observed in the univariate version:

- For the Longely data, even small values of k have substantial impact on $||\boldsymbol{\beta}_k||$
- Even more dramatic is the effect on the size of the confidence ellipsoids
- Shrinkage in variance (e.g., $|\boldsymbol{\Sigma}_k|^{1/p}$) tends to be in the same direction as shrinkage in coefficients
- The bivariate path of shrinkage in $\boldsymbol{\beta}_k$ is often, but not always monotonic
  - e.g., β_{GNP} vs. β_{Pop}
  - All bivariate paths for Population and GNP.deflator
- The covariance between pairs of coefficients (orientation of ellipses) also tends to change systematically, but not always.

The scatterplot matrix format makes it particularly easy to see the effects on bias and variance for a given variable.

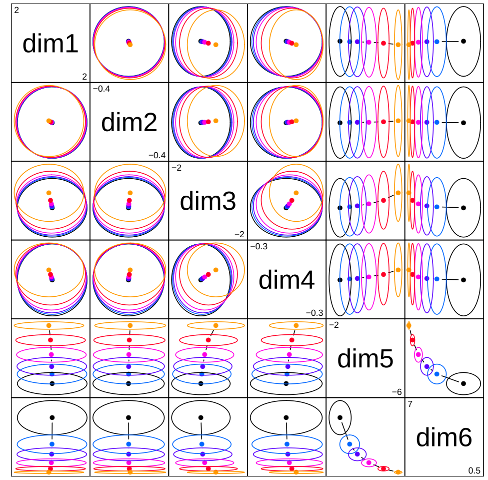
3D ridge trace plots: plot3d() method

```
> plot3d(lridge, radius=0.5)
```



Ridge trace plots in PCA / SVD space: pca method()

```
> plridge <- pca.ridge(lridge)
> pairs(plridge, col=clr, radius=0.5, diag.cex=3)
```

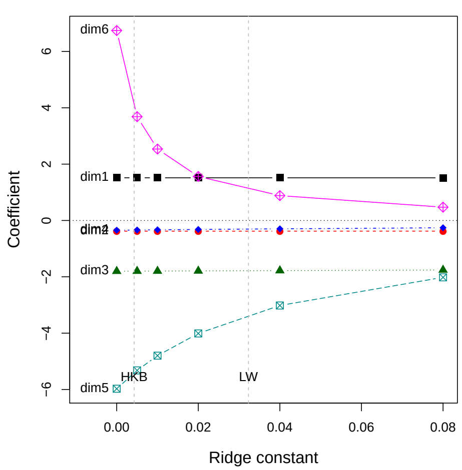


- The ellipsoids are rotated to the principal axes of $\mathbf{X}^\mathsf{T}\mathbf{X}$
- SVD of $\mathbf{X} = \mathbf{U}\mathbf{D}\mathbf{V}^\mathsf{T}$ implies: $\mathcal{E}(\beta, \Sigma) \mapsto \mathcal{E}(\mathbf{V}\beta, \mathbf{V}^\mathsf{T}\Sigma\mathbf{V})$
- Transformed ellipsoids have their major/minor axes aligned with coordinate axes.
- It is easy to see that shrinkage occurs only in the space of the smallest eigenvalues

Ridge trace plots in PCA / SVD space: pca method()

We can also see this in the univariate trace plot in the transformed PCA/SVD space

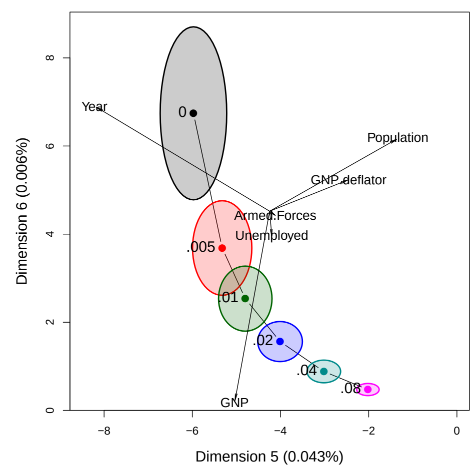
```
> traceplot(plridge)
```



- Essentially no shrinkage in Dim 1–Dim 4
- Dim 5 and Dim 6 are shrunk towards 0
- Greater shrinkage for the smallest dimension: Dim 6

View in PCA space of smallest dimensions: biplot() method

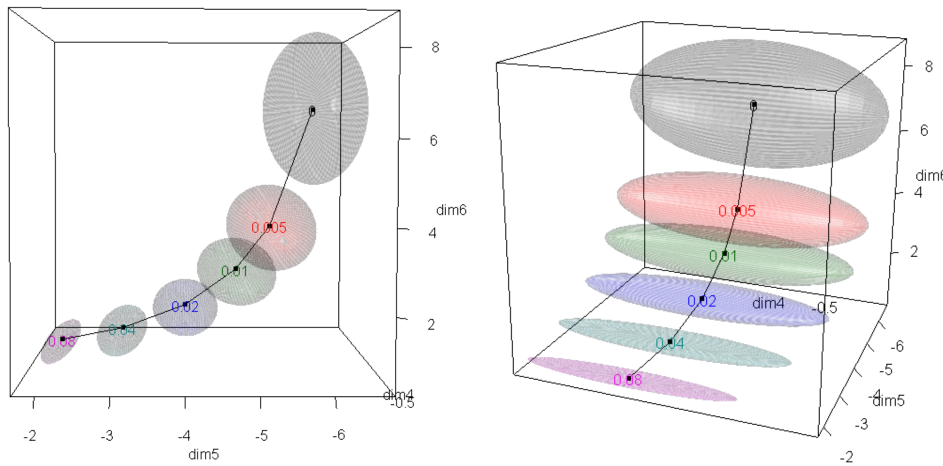
```
> biplot(plridge, col=clr, radius=.5, cex.lab=1.25, prefix="Dimension ")
```



- View the variance ellipsoids in the space of the smallest dimensions
- This is where the greatest shrinkage takes place!
- Variable vectors show how these dimensions relate to the original variables ["biplot"]
- GNP, Year & Pop contribute most to Dim 6

3D views in PCA space

```
> plot3d.ridge(plridge, variables=4:6, radius=.5)
```



VIFs in PCA/SVD space

Finally, note that the transformation to PCA space makes all transformed predictors orthogonal, so the VIFs are all 1.0

```
> vif(plridge)
```

	dim1	dim2	dim3	dim4	dim5	dim6
0.000	1	1	1	1	1	1
0.005	1	1	1	1	1	1
0.010	1	1	1	1	1	1
0.020	1	1	1	1	1	1
0.040	1	1	1	1	1	1
0.080	1	1	1	1	1	1

Added benefit: biplot views help to make the results of PCA regression more interpretable

Multivariate bootstrap methods

If normal theory too restrictive, or if there is no closed-form expression for $\widehat{\boldsymbol{\Sigma}}$ simple non-parametric versions can be calculated via bootstrap methods as follows:

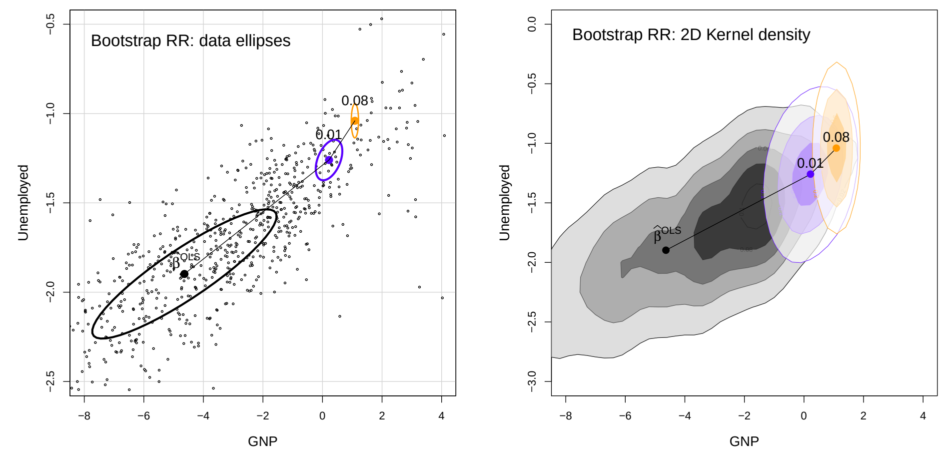
- Generate B bootstrap estimates $\widetilde{\beta}_k^b$, $b = 1, 2, \ldots, B$, each $p \times 1$, by resampling from the rows of available data, $(\mathbf{y}, \mathbf{X})$.
  - For given k, the bootstrap estimate $\widetilde{\beta}_k = \text{Ave}(\widetilde{\beta}_k^b) = B^{-1}\Sigma_b \widetilde{\beta}_k^b$.
  - Bootstrap estimate of $\widetilde{\boldsymbol{\Sigma}}_k$ can be computed as the empirical covariance matrix of $\widetilde{\beta}_k^b$,

$$\widetilde{\boldsymbol{\Sigma}}_k = B^{-1} \sum_{b=1}^{B} (\widetilde{\beta}_k^b - \widetilde{\beta}_k)(\widetilde{\beta}_k^b - \widetilde{\beta}_k)^{\mathsf{T}} \ . \tag{8}$$

- Simple display: data ellipsoids of the bootstrap sample estimates, $\widetilde{\beta}_k \oplus \widetilde{\boldsymbol{\Sigma}}_k^{1/2} \mathcal{S}$. [Still assumes normality of bootstrap estimates.]
- Alternatively, use non-parametric density estimation → smoothed approximations to the joint distribution of the $\widetilde{\beta}_k^b$ (only in 2D)

Multivariate bootstrap methods

Results for $B = 800$ bootstrap samples of the ridge regression estimates for GNP and Unemployed in Longley's data.



Conjecture: RR shrinkage, in addition to increasing precision, also improves the normal approximation on which these graphical methods rely.

Summary and conclusions I

- Shrinkage in ridge regression & related methods is a multivariate problem
  - requires simultaneous visualization of "bias" ($||\beta_k||$) and precision ($|\mathbf{\Sigma}_k|^{-1/p}$)
  - this is achieved by 2D and 3D plotting methods displaying the covariance ellipsoids of the ridge estimates, $\mathcal{E}(\widehat{\beta}_k, \widehat{\mathbf{\Sigma}}_k)$
- Even static, 2D views, e.g., `pairs()` plots, can be far more revealing than univariate ridge trace plots
  - Ellipsoid centers ($\widehat{\beta}_k$) show how parameter estimates shrink jointly
  - Ellipsoid size and shape ($\widehat{\mathbf{\Sigma}}_k$) show how parameter variances and covariances shrink jointly

Summary and conclusions II

- Higher-p problems are more easily visualized by transforming the ellipsoids $\mathcal{E}(\widehat{\beta}_k, \widehat{\mathbf{\Sigma}}_k)$ to PCA/SVD space, $\mathcal{E}(\mathbf{V}\widehat{\beta}_k, \mathbf{V}^\mathsf{T}\widehat{\mathbf{\Sigma}}_k\mathbf{V})$
  - The dimensions corresponding to the smallest singular values provide the most informative views of shrinkage
  - Interpretation in terms of the original variables is facilitated by plotting projections of variable vectors in this space ["biplot"]
- Extensions:
  - The same graphical ideas apply to any shrinkage/selection method that provides estimates $\widehat{\beta}_k$ (a `coef()` method) and variance-covariance estimates $\widehat{\mathbf{\Sigma}}_k$ (a `vcov()` method).
  - When variance-covariance estimates are unavailable analytically, they can be approximated by bootstrap methods.

Summary and conclusions III

- Graphical inspiration:
  - This paper arose as one example of the idea that multivariate views of data are illuminated by the geometry of ellipsoids

"Once you tune in to ellipses you will begin to see them everywhere."

– FIN –