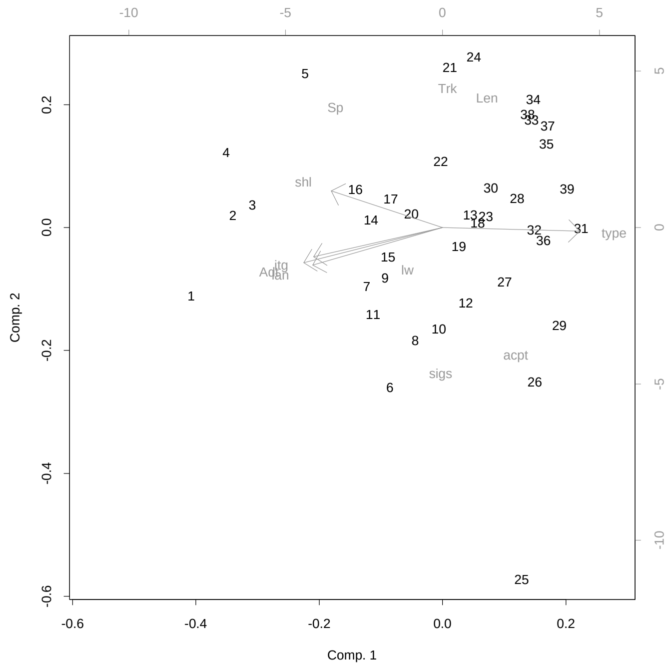

Principal Components Regression

The *principal components* of Z are the columns of the matrix

$$F = ZA$$

Since $AA^T = I$, Equation (2) can be written

$$\begin{aligned} \underline{y} &= \mathbf{1}\beta_0 + Z\alpha + \epsilon \\ &= \mathbf{1}\beta_0 + (ZA)(A^T\alpha) + \epsilon \\ &= \mathbf{1}\beta_0 + F\gamma + \epsilon \end{aligned} \tag{5}$$

(where $\gamma = A^T\alpha$), and this is equivalent to our original Equation (1).

The least-squares estimates of γ are $\hat{\gamma} = (F^TF)^{-1}F^Ty$, and as usual

$$\mathrm{Var}(\hat{\gamma}) = \sigma_\epsilon^2(F^TF)^{-1} = \sigma_\epsilon^2(A^TZ^TZA)^{-1} = \frac{\sigma_\epsilon^2}{n-1}\Lambda^{-1}$$

is a diagonal matrix, and $\mathrm{Var}(\hat{\gamma}_k) = \frac{1}{n-1}\sigma^2\lambda_k^{-1}$, so that

- the $\hat{\gamma}$'s are uncorrelated; and
- the data tells us the least about the γ_k for the smallest eigenvalue λ_k.

This suggests that we can "improve" the model by dropping columns of F corresponding to high Var($\hat{\gamma}_k$) i.e. low λ_j. If the original model in Equation (1) was the correct model for the data, then the estimates $\hat{\beta}$, $\hat{\alpha}$ and $\hat{\gamma}$ in these problems should all be unbiased.

To see what the effect of dropping columns of F is, consider minimizing the sum of mean square errors (SMSE) in the coefficients, instead of in the predicted values,

$$\begin{aligned} SMSE &= \sum_{j=1}^{k} E[(\hat{\gamma}_j - \gamma_j)^2] \\ &= \sum_{j=1}^{k} \{\mathrm{Var}(\hat{\gamma}_j) + [bias(\hat{\gamma}_j)]^2\} \\ &= \frac{\sigma_\epsilon^2}{n-1} \sum_{j=1}^{k} \lambda_j^{-1} + \sum_{j=1}^{k} \{E[\hat{\gamma}_i - \gamma_j]\}^2 \end{aligned} \tag{6}$$

where $\hat{\gamma}_j$ are the least-squares estimates. When the full set of least-squares estimates are used, $\hat{\gamma}_j$ will be unbiased and the "bias" term on the right will be zero, so

$$SMSE_{OLS} = \frac{\sigma_\epsilon^2}{n-1} \sum_{j=1}^{k} \lambda_j^{-1}$$

Because the columns of W are orthogonal, when we drop columns we do not change the least-squares estimates of the remaining coefficients. Dropping the last m columns for example we obtain

$$SMSE_{PC(k-m)} = \frac{\sigma_\epsilon^2}{n-1} \sum_{j=1}^{k-m} \lambda_j^{-1} + \sum_{j=k-m+1}^{k} \gamma_j^2$$

This is a classic case of the *bias/variance tradeoff* in statistics: by dropping only a few terms we can reduce SMSE by paying for a large drop in variance with a small increase in bias; clearly we can minimize $SMSE_{PC(k-m)}$ over m if desired.

For the highway data that we considered last time, we can estmate both SMSE's:

```
> names(pc.cor)
 [1] "sdev"         "loadings"     "correlations" "scores"       "center"
 [6] "scale"        "n.obs"        "call"         "factor.sdev"  "coef"
```

```
> summary(lm(y ~ pc.cor$scores))
```

```
Coefficients:
                          Value Std. Error  t value Pr(>|t|)
          (Intercept)    3.9333   0.1876    20.9661   0.0000
 pc.cor$scoresComp. 1    0.2711   0.0933     2.9060   0.0072
 pc.cor$scoresComp. 2   -0.9628   0.1158    -8.3139   0.0000
 pc.cor$scoresComp. 3   -0.2500   0.1762    -1.4185   0.1675
 pc.cor$scoresComp. 4   -0.0058   0.1997    -0.0290   0.9771
 pc.cor$scoresComp. 5   -0.3396   0.2394    -1.4188   0.1674
 pc.cor$scoresComp. 6   -0.1522   0.2508    -0.6068   0.5490
 pc.cor$scoresComp. 7    0.0724   0.2882     0.2510   0.8037
 pc.cor$scoresComp. 8    0.1167   0.3430     0.3402   0.7364
 pc.cor$scoresComp. 9    0.0296   0.3838     0.0772   0.9391
pc.cor$scoresComp. 10   -0.0931   0.5475    -0.1700   0.8663
pc.cor$scoresComp. 11    0.1133   0.7480     0.1515   0.8807

Residual standard error: 1.172 on 27 degrees of freedom
Multiple R-Squared: 0.7527
F-statistic: 7.472 on 11 and 27 degrees of freedom,
the p-value is 1.051e-05

> dim(high)
[1] 39 12
> n _ 39

> print(SMSE.OLS _ (1.172)^2 * sum(pc.cor$factor.sdev^(-2))/(n-1) )
[1] 1.458874
> coefs _ lm(y ~ pc.cor$scores)$coef
> coefs[-c(1:3)]
 pc.cor$scoresComp. 3 pc.cor$scoresComp. 4 pc.cor$scoresComp. 5
           -0.2499623         -0.005792799            -0.339593

 pc.cor$scoresComp. 6 pc.cor$scoresComp. 7 pc.cor$scoresComp. 8
           -0.1521965           0.07235734             0.116677

 pc.cor$scoresComp. 9 pc.cor$scoresComp. 10 pc.cor$scoresComp. 11
           0.02961616           -0.09308246             0.1133036
> print( SMSE.PC2 _ (1.172)^2 * sum(pc.cor$factor.sdev[1:2]^(-2))/(n-1) +
+ sum(coefs[-c(1:3)]^2) )
[1] 0.2649427
```

So we win the bias/variance tradeoff game by regressing on only the first two principal components (which we named, in our previous analysis, the `major.road` and `slow.downs` indices).