Linear Regression Cont.

Thomas Lonon

Division of Financial Engineering
Stevens Institute of Technology

September 20, 2017

Let X^T be an input vector given by:

$$X^T = (X_1, X_2, \ldots, X_p)$$

to predict a real-valued output Y. In this form the linear regression model is given by:

$$f(X) = \beta_0 + \sum_{j=1}^{p} X_j \beta_j$$

This model assumes that the regression function $\mathbb{E}[Y|X]$ is linear (or is a reasonable approximation)

Typically we have a set of training data $(x_1, y_1) \ldots (x_N, y_N)$ to estimate the parameters β. Each $x_i = (x_{i1}, x_{i2}, \ldots, x_{ip})^T$ is a vector of predictors of the i^{th} data point. For *least squares* estimation method we minimize the RSS:

$$RSS(\beta) = \sum_{i=1}^{N} (y_i - f(x_i))^2$$

$$= \sum_{i=1}^{N} (y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j)^2$$

Denote by $\mathbf{X}$ the $N \times (p+1)$ matrix with each row an input vector with a 1 in the first column and let $\mathbf{y}$ be the N vector of outputs. Using these definitions we have:

$$RSS(\beta) = (\mathbf{y} - \mathbf{X}\beta)^T(\mathbf{y} - \mathbf{X}\beta)$$

which is a quadratic function of the $p+1$ parameters.

Differentiating with respect to β gives us:

$$\frac{\partial RSS}{\partial \beta} = -2\mathbf{X}^T(\mathbf{y} - \mathbf{X}\beta)$$
$$\frac{\partial^2 RSS}{\partial \beta \partial \beta^T} = 2\mathbf{X}^T\mathbf{X}$$

If we assume that $\mathbf{X}$ has full column rank, and so $\mathbf{X}^T\mathbf{X}$ is positive definite, then we set the first derivative to 0:

$$\mathbf{X}^T(\mathbf{y} - \mathbf{X}\beta) = 0$$

which gives us the solution:

$$\hat{\beta} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$$

and so the fitted values are:

$$\hat{\mathbf{y}} = \mathbf{X}\hat{\beta} = \mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$$

We make the assumptions: y_i are uncorrelated and have constant variance σ^2 and that the x_i are fixed. The covariance matrix of the estimated parameters are given by:

$$Var(\hat{\beta}) = (\mathbf{X}^T\mathbf{X})^{-1}\sigma^2$$

with σ^2 estimated by:

$$\hat{\sigma}^2 = \frac{1}{N-p-1}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2$$

We also assume that the deviation of Y around its mean are additive and Gaussian:

$$\begin{aligned} Y &= \mathbb{E}[Y|X_1, \ldots, X_p] + \varepsilon \\ &= \beta_0 + \sum_{j=1}^{p} X_j \beta_j + \varepsilon \end{aligned}$$

with $\varepsilon \sim N(0, \sigma^2)$ and so

$$\hat{\beta} \sim N(\beta, (\mathbf{X}^T\mathbf{X})^{-1}\sigma^2)$$

$$(N - p - 1)\hat{\sigma}^2 \sim \sigma^2 \chi^2_{N-p-1}$$

Multiple Outputs

To predict multiple outputs $(Y_1, \ldots, Y_K)$ from inputs $(X_0, \ldots, X_p)$, we assume a linear model for each:

$$Y_k = \beta_{0k} + \sum_{j=1}^{p} X_j \beta_{jk} + \varepsilon_k$$
$$= f_k(X) + \varepsilon_k$$

Assuming N training cases, we write the model in matrix notation as:

$$\mathbf{Y} = \mathbf{XB} + \mathbf{E}$$

[2]

$$RSS(\mathbf{B}) = \sum_{k=1}^{K}\sum_{i=1}^{N}(y_{ik} - f_k(x_i))^2$$
$$= tr[(\mathbf{Y} - \mathbf{XB})^T(\mathbf{Y} - \mathbf{XB})]$$

which results in the same least-squares estimates as before:

$$\hat{\mathbf{B}} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{Y}$$

If the errors $\varepsilon = (\varepsilon_1, \ldots, \varepsilon_K)$ are correlated, then we modify to favor multivariate regression.

For $Cov(\varepsilon) = \Sigma$, we have

$$RSS(\mathbf{B}, \Sigma) = \sum_{i=1}^{N} (y_i - f(x_i))^T \Sigma^{-1} (y_i - f(x_i))$$

There are two reasons why the least square estimates might not be "good enough"

- Prediction Accuracy - least squares estimates have low bias, but high variance
- Interpretation - If we have a large number of predictors, we would ideally like to determine a smaller subset

So we utilize variations on this approach to fit subsets of the predictors.

Algorithm 6.1: Best Subset Selection:

1. Let $\mathcal{M}_0$ denote the *null model*, which contains no predictors. This model simply predicts the sample mean for each observation.
2. For $k = 1, 2, \ldots, p$:
   - 2.1 Fit all $\binom{p}{k}$ models that contain exactly k predictors
   - 2.2 Pick the best among these $\binom{p}{k}$ models, and call it $\mathcal{M}_k$. Here *best* is defined as having the smallest RSS, or equivalently largest R^2
3. Select a single best model from among $\mathcal{M}_0, \mathcal{M}_1, \ldots, \mathcal{M}_p$ using cross-validated prediction error, C_p, AIC, BIC, or adjusted R^2

[1]

Algorithm 6.2: Forward Stepwise Selection:

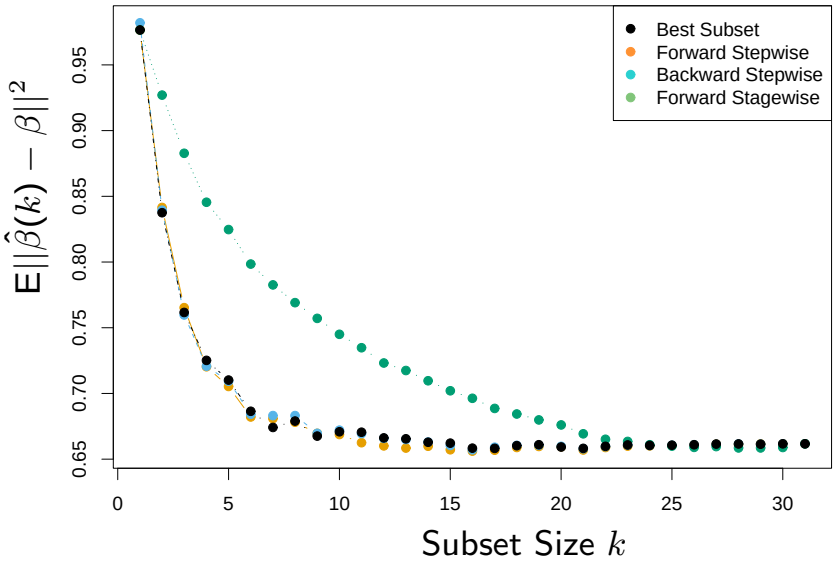
1. Let $\mathcal{M}_0$ denote the *null model*, which contains no predictors.
2. For $k = 0, 1, \ldots, p-1$:
   - 2.1 Consider all $p - k$ models that augment the predictors in $\mathcal{M}_k$ with one additional predictor
   - 2.2 Choose the *best* among these $p - k$ models, and call it $\mathcal{M}_{k+1}$. Here *best* is defined as having the smallest RSS, or equivalently largest R^2
3. Select a single best model from among $\mathcal{M}_0, \mathcal{M}_1, \ldots, \mathcal{M}_p$ using cross-validated prediction error, C_p, AIC, BIC, or adjusted R^2

[1]

Algorithm 6.3: Backward Stepwise Selection:

1. Let $\mathcal{M}_p$ denote the *full* model, which contains all p predictors.
2. For $k = p, p-1, \ldots, 1$:
   - 2.1 Consider all k models that contain all but one of the predictors in $\mathcal{M}_k$, for a total of $k-1$ predictors.
   - 2.2 Choose the *best* among these k models, and call it $\mathcal{M}_{k-1}$. Here *best* is defined as having the smallest RSS, or equivalently largest R^2
3. Select a single best model from among $\mathcal{M}_0, \mathcal{M}_1, \ldots, \mathcal{M}_p$ using cross-validated prediction error, C_p, AIC, BIC, or adjusted R^2

[1]



[2]

FIGURE 3.6. *Comparison of four subset-selection techniques on a simulated linear regression problem* $Y = X^T\beta + \varepsilon$. *There are* $N = 300$ *observations on* $p = 31$ *standard Gaussian variables, with pairwise correlations all equal to* 0.85. *For* 10 *of the variables, the coefficients are drawn at random from a* $N(0, 0.4)$ *distribution; the rest are zero. The noise* $\varepsilon \sim N(0, 6.25)$, *resulting in a signal-to-noise ratio of* 0.64. *Results are averaged over* 50 *simulations. Shown is the mean-squared error of the estimated coefficient* $\hat{\beta}(k)$ *at each step from the true* β.

Mallow's C_p

For a fitted least squares model containing d predictors, the C_p estimate of test MSE is computed using the equation:

$$C_p = \frac{1}{n}(RSS + 2d\hat{\sigma}^2)$$

where $\hat{\sigma}^2$ is an estimate of the variance of the error

Mallow's C_p is sometimes defined as

$$C'_p = \frac{RSS}{\hat{\sigma}^2} + 2d - n$$

The model with the smallest C_p will also have the smallest C'_p

Akaike Information Criterion (AIC)

For least square models:

$$AIC = \frac{1}{n\hat{\sigma}^2}(RSS + 2d\hat{\sigma}^2)$$

(an additive constant has been omitted for simplicity)

The AIC can be defined more generally in terms of log-likelihood as

$$AIC = -2\log L + 2d$$

Bayesian Information Criterion (BIC)

For the least squares model with d predictors up to irrelevant constants

$$BIC = \frac{1}{n}(RSS + d\hat{\sigma}^2 \log n)$$

Note that BIC replaces the $2d\hat{\sigma}^2$ with $d\hat{\sigma}^2 \log n$.

Since $\log n > 2$ for $n > 7$, the BIC places a heavier penalty on models with many variables

Adjusted R^2

Recall:

$$R^2 = 1 - \frac{RSS}{TSS}$$

Note that this will always choose all of the variables

Instead to create a "punishment" for including noise terms we define the adjusted R^2 as:

$$R^2_{adj} = 1 - \frac{RSS/(n-d-1)}{RSS/(n-1)}$$

Ridge Regression

Previously defined we have the least squares fitting procedure that estimates $\beta_0, \beta_1, \ldots, \beta_p$ and minimizes

$$RSS = \sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2$$

Ridge Regression is similar to least squares, but it finds the estimates that minimize:

$$\sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 + \lambda \sum_{j=1}^{p} \beta_j^2 = RSS + \lambda \sum_{j=1}^{p} \beta_j^2$$

where $\lambda \geq 0$ is a *tuning parameter*

This ridge regression can be alternately formulated as either:

$$\hat{\beta}^{ridge} = \arg\min_{\beta} \sum_{i=1}^{N} \left(y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2$$

subject to $\sum_{j=1}^{p} \beta_j^2 \leq t$

or in matrix notation:

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

$$\hat{\beta}^{ridge} = (\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^T\mathbf{y}$$

As with least squares, ridge regression seeks coefficient estimates that fit the data well

However, the second term $\lambda \sum \beta_j^2$ (which is called the *shrinkage penalty*) is small when $\beta_0, \ldots, \beta_p$ are close to 0

For each λ you get a different set of parameters $\hat{\beta}_\lambda^R$.

The standard least squares coefficient estimates are **scale equivariant**

This ridge regression is not scale equivariant so we apply the ridge regression after **standardizing the predictors**

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

[1]

The Lasso

A relatively recent alternative to ridge regression that picks the coefficients $\hat{\beta}_\lambda^L$ that minimizes:

$$\sum_{i=1}^{n}\left(y_i - \beta_0 - \sum_{j=1}^{p}\beta_j x_{ij}\right)^2 + \lambda\sum_{j=1}^{p}|\beta_j| = RSS + \lambda\sum_{j=1}^{p}|\beta_j|$$

This approach produces **sparse** models.

Alternate Formulation

$$\min_{\beta} \left\{ \sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 \right\}, \text{ subject to } \sum_{j=1}^{p} |\beta_j| \leq s$$

$$\min_{\beta} \left\{ \sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 \right\}, \text{ subject to } \sum_{j=1}^{p} \beta_j^2 \leq s$$

[2]

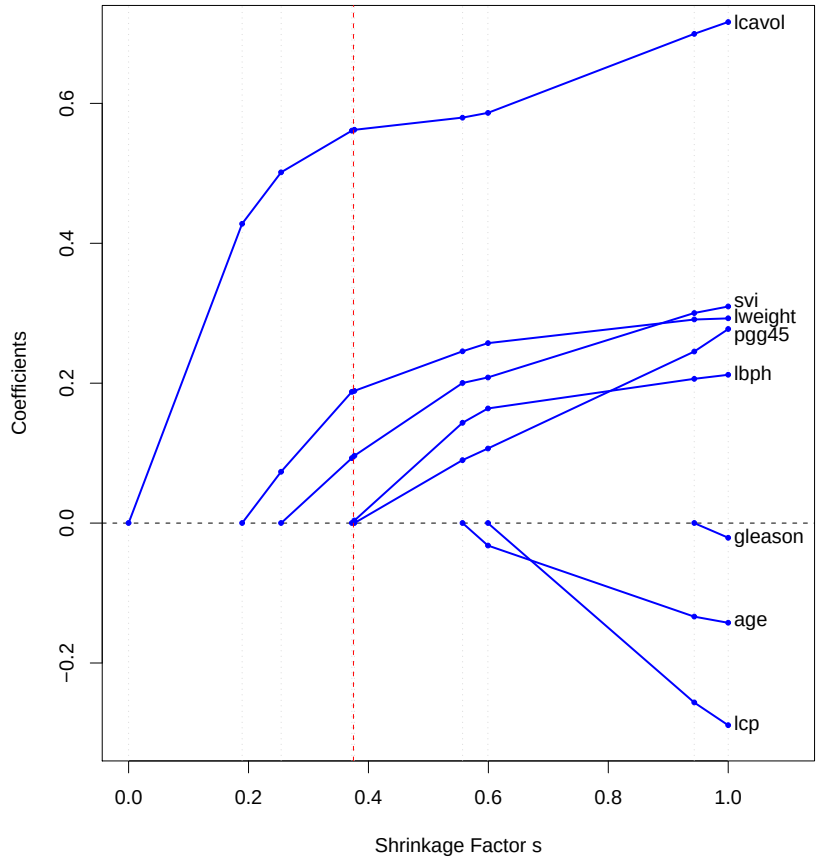


FIGURE 3.10. *Pro les of lasso coefficients, as the tuning parameter t is varied. Coefficients are plotted versus* $s = t/\sum_1^p |\hat{\beta}_j|$. *A vertical line is drawn at* $s = 0.36$, *the value chosen by cross-validation. Compare Figure 3.8 on page 9; the lasso pro les hit zero, while those for ridge do not. The pro les are piece-wise linear, and so are computed only at the points displayed;*

Least Angle Regression

Algorithm 3.2: Least Angle Regression

1. Standardize the predictors to have mean zero and unit norm. Start with the residual $\mathbf{r} = \mathbf{y} - \bar{\mathbf{y}}$, $\beta_0, \beta_1, \ldots, \beta_p = 0$.
2. Find the predictor $\mathbf{x}_j$ most correlated with $\mathbf{r}$
3. Move β_j from 0 towards its least-squares coefficient $\langle \mathbf{x}_j, \mathbf{r} \rangle$, until some other competitor $\mathbf{x}_k$ has as much correlation with the current residual as does $\mathbf{x}_j$.
4. Move β_j and β_k in the direction defined by their joint least squares coefficient of the current residual on $(\mathbf{x}_j, \mathbf{x}_k)$, until some other competitor $\mathbf{x}_\ell$ has as much correlation with the current residual.
5. Continue in this way until all p predictors have been entered. After $\min(N - 1, p)$ steps, we arrive at the full least-squares solution.

[2]

Let $Z_1, \ldots, Z_M$ represent $M < p$ **linear combinations** of the original p predictors

$$Z_m = \sum_{j=1}^{p} \phi_{jm} X_j$$

for constants $\phi_{1m}, \ldots, \phi_{pm}$, $m = 1, \ldots, M$.

We then fit the linear regression model using least squares

$$y_i = \theta_0 + \sum_{m=1}^{M} \theta_m z_{im} + \epsilon_i, i = 1, \ldots, n$$

These techniques are referred to as **dimension reduction**.

Note that before the reduction we estimate $p+1$ coefficients $(\beta_0, \ldots, \beta_p)$ whereas after the reduction we are fitting $M+1$ with $M < p$ $(\theta_0, \ldots, \theta_M)$

$$\sum_{m=1}^{M} \theta_m z_{im} = \sum_{m=1}^{M} \theta_m \sum_{j=1}^{p} \phi_{jm} x_{ij} = \sum_{j=1}^{p} \sum_{m=1}^{m} \theta_m \phi_{jm} x_{ij} = \sum_{j=1}^{p} \beta_j x_{ij}$$

$$\beta_j = \sum_{m=1}^{M} \theta_m \phi_{jm}$$

Principal Components Analysis

A technique for reducing the dimension of a $n \times p$ data matrix $\mathbf{X}$

The **first principal component** direction of the data is that along which the observations vary the most[1]

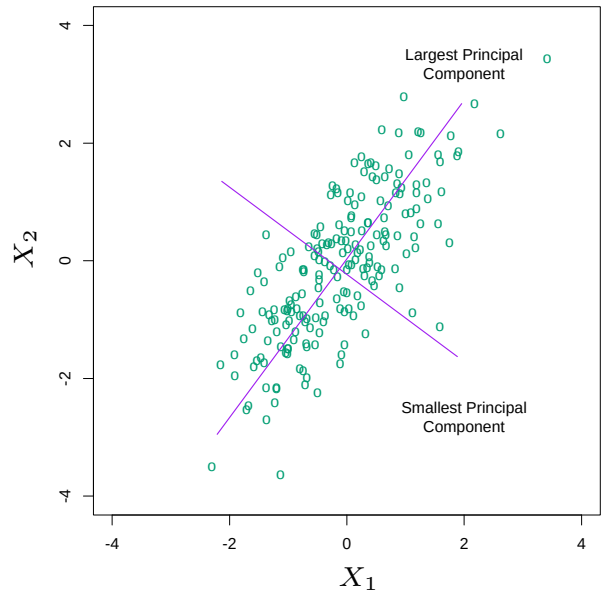
The values of $z_{11}, \ldots, z_{n1}$ are known as the **principal component scores**

Principal Components Regression

This involves constructing the first M principal components $(Z_1, \ldots, Z_m)$ and using them as predictors in a linear regression model fit using least squares

We assume that "the directions in which $X_1, \ldots, X_p$ show the most variation are the directions that are associated with Y"[1]

If the assumption holds, then fitting the least square model to $Z_1, \ldots, Z_m$ will result in a better fit than $X_1, \ldots, X_p$



[2]

FIGURE 3.9. *Principal components of some input data points. The largest principal component is the direction that maximizes the variance of the projected data, and the smallest principal component minimizes that variance. Ridge regression projects* $\mathbf{y}$ *onto these components, and then shrinks the coefficients of the low-variance components more than the high-variance components.*

Partial Least Squares

After standardizing the p predictors, compute the first direction Z_1 by setting each ϕ_{j1} equal to the coefficient from the linear regression of Y onto X_j

Next we adjust each of the variables for Z_1 by regressing each variable on Z_1 and taking residuals. Then compute Z_2 using this orthogonalized data. Repeating this approach up to Z_M

Note that this approach typically doesn't outperform ridge regression or PCR

Algorithm 3.3: Partial Least Squares

1. Standardize each $\mathbf{x}_j$ to have mean zero and variance one. Set $\hat{\mathbf{y}}^{(0)} = \bar{y}\mathbf{1}$, and $\mathbf{x}_j^{(0)} = \mathbf{x}_j, j = 1, \ldots, p$.
2. For $m = 1, 2, \ldots, p$
   - 2.1 $\mathbf{z}_m = \sum_{j=1}^{p} \hat{\varphi}_{mj}\mathbf{x}_j^{(m-1)}$, where $\hat{\varphi}_{mj} = \langle \mathbf{x}_j^{(m-1)}, \mathbf{y} \rangle$
   - 2.2 $\hat{\theta}_m = \langle \mathbf{z}_m, \mathbf{y} \rangle / \langle \mathbf{z}_m, \mathbf{z}_m \rangle$
   - 2.3 $\hat{\mathbf{y}}^{(m)} = \hat{\mathbf{y}}^{(m-1)} + \hat{\theta}_m \mathbf{z}_m$
   - 2.4 Orthogonalize each $\mathbf{x}_j^{(m-1)}$ with respect to $\mathbf{z}_m$:
     $\mathbf{x}_j^{(m)} = \mathbf{x}_j^{(m-1)} - [\langle \mathbf{z}_m, \mathbf{x}_j^{(m-1)} \rangle / \langle \mathbf{z}_m, \mathbf{z}_m \rangle]\mathbf{z}_m, j = 1, 2, \ldots, p$
3. Output the sequence of fitted vectors $\{\hat{\mathbf{y}}^{(m)}\}_1^p$. Since the $\{\mathbf{z}_\ell\}_1^m$ are linear in the original $\mathbf{x}_j$, so is $\hat{\mathbf{y}}^{(m)} = \mathbf{X}\hat{\beta}^{pls}(m)$. These linear coefficients can be recovered from the sequence of PLS transformations.

Removing the Additive

In the linear model, when we look at the effect on Sales by TV and Radio, we assume it is governed by the model:

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \varepsilon$$

We can introduce an **interaction term** to this model for synergistic effects:

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_1 X_2 + \varepsilon$$

Non-linear Relationships

We can also expand the model to include non-linear terms. For example we could fit the model:

$$Y = \beta_0 + \beta_1 X + \beta_2 X^2 + \varepsilon$$

where Y is a function of X. This could be expanded for multiple predictors and any degree of each predictor.

Potential Problems

1. Non-linearity of the response-predictor relationships
2. Correlation of error terms
3. Non-constant variance of error terms
4. Outliers
5. High-leverage points
6. Collinearity

[1]

Non-linearity of the response-predictor relationships: We can check the fit of the linear model with *residual plots*. These plot the residuals, $e_i = y_i - \hat{y}_i$, versus the predictor x_i (or in the case of multiple regression versus $\hat{y}_i$)

Correlation of error terms: If the error terms are correlated (i.e. knowing ε_i gives you information about ε_{i+1}) then linear regression will not work. You would in that case need to use an alternate approach such as time series (definitely not covered in this course)

Non-constant variance of error terms: For nonconstant variance terms, the *heteroscedasticity* can be observed in a funnel shape of the residual terms.

Outliers: Data points that have the the actual value y_i far from the predicted value $\hat{y}_i$ are considered outliers. These data points when included do not necessarily change the linear regression fit much, but they do cause problems in measures such as the RSE and R^2

High-leverage points: Points in which the predictor x_i is unusual are considered *high leverage*. We identify these using the *leverage statistic* given by

$$h_i = \frac{1}{n} + \frac{(x_i - \bar{x})^2}{\sum_{i'=1}^{n}(x_{i'} - \bar{x})^2}$$

If this values greatly exceeds $(p+1)/n$ then we suspect this has high leverage

Collinearity: A way to assess *multicollinearity* is by computing the *variance inflation factor* (VIF) by:

$$VIF(\hat{\beta}_j) = \frac{1}{1 - R^2_{X_j|X_{-j}}}$$

where $R^2_{X_j|X_{-j}}$ is the R^2 regression of X_j onto all other predictors.

[1] Trevor Hastie Gareth James, Daniela Witten and Robert Tibshirani. *An Introduction to Statistical Learning with Applications in R*. Number v. 6. Springer, 2013.

[2] Robert Tibshirani Trevor Hastie and Jerome Friedman. *The Elements of Stastical Learning: Data Mining, Inference, and Prediction*. Number v.2 in Springer Series in Statistics. Springer, 2009.