# Towards Modern Regression

Daniel Lawson University of Bristol

Lecture 02.1 (v2.0.0)

# Signposting

- This lecture covers:
  - Classical regression
  - Towards Modern Regression - the vectorised version, which uses Matrix algebra.
  - Leave-one-out Cross Validation
- The maths here underpins almost all modern data science.

# Questions

- What is Regression (not) for?
- What role does Matrix Algebra have in advanced Machine Learning?
- How do you know one model is “better” than another?

# Before we start: Vector Notation

- There are several choices of convention that we have to make
- Vectors of length $k$ are also matrices, but are they $k \times 1$ or $1 \times k$?
- We use $k \times 1$, i.e. column vectors
- Similarly there are choices about matrix derivatives
- We use derivative with respect to a column vector as a row vector
- Some resources differ and have everything transposed as a consequence

# Covariance

- *A reminder: understanding covariance and correlation is a prerequisite*
- covariance is simply the second (central) moment:

$$\operatorname{cov}(X, Y) = \mathbb{E}\left[(X - \mathbb{E}[X])(Y - \mathbb{E}[Y])\right]$$

- Recall that we typically use **unbiased** estimators which often slightly different from natural theoretical analogue. The **sample covariance** is:

$$\operatorname{cov}(X, Y) = \frac{1}{N-1} \sum_{i=1}^{N} (X_i - \bar{X})(Y_i - \bar{Y})$$

# Linear algebra view of covariance

- The **covariance matrix** of a random variable $\mathrm{X}$
- Where $\mathrm{X}$ is a vector-valued RV with length $k$,
- has entries:

$$\mathrm{Cov}(\mathrm{X})_{ij} = \mathrm{Cov}(\mathrm{X}_i, \mathrm{X}_j) = \mathbb{E}[(\mathrm{X}_i - \mu_i)(\mathrm{X}_j - \mu_j)].$$

- The matrix form for this is:

$$\Sigma = \mathbb{E}[(\mathrm{X} - \mathbb{E}[\mathrm{X}])(\mathrm{X} - \mathbb{E}[\mathrm{X}])^T],$$

- Where $\mu = \mathbb{E}[\mathrm{X}]$.

# Correlation

- Correlation is simply a **normalised measure** of covariance.

$$\rho_{X,Y} = \frac{\operatorname{cov}(X,Y)}{\sigma_X \sigma_Y}$$

- It takes values between -1 and 1.
- Sample correlation uses the unbiased estimator of covariance, to account for the number of degrees of freedom in the data.
- **Question**: What should we (not) take the correlation of?
  - See rank correlation, canonical correlation, etc.

# Linear algebra view of correlation

- Division by standard deviations is required to correctly generalise the **scalar correlation**:

$$\mathrm{Corr}(X, Y) = \frac{\mathbb{E}[(X - \mu_X)(Y - \mu_Y)]}{\sigma_X \sigma_Y}.$$

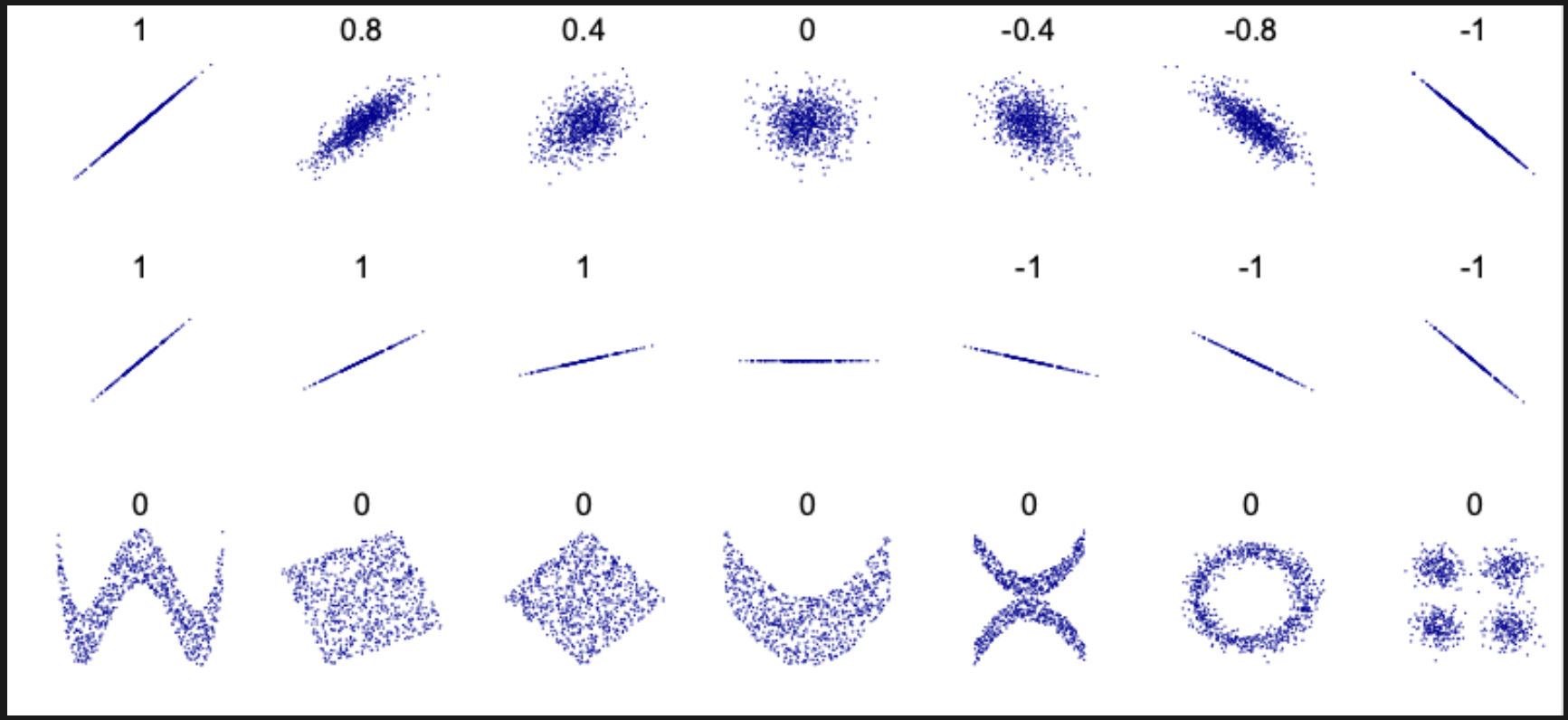
- The **matrix form** for correlation is:

$$\mathrm{Corr}(\mathrm{X}) = (\mathrm{diag}(\Sigma))^{-1/2}\, \Sigma\, (\mathrm{diag}(\Sigma))^{-1/2}$$

- The matrix inversion is not computationally challenging because it is for a **diagonal matrix**.

# Examples

From Wikipedia: Correlation_and_dependence



# Regression

- **Regression**, considers the relationship of a response variable as determined by one or more explanatory variables.
  - Regression is designed to help **make predictions** of $y$ when we observe $x$.
  - It is a **conditional model**, and **not** a joint model of $x$ and $y$. This is its **strength**.
  - It predicts the **best guess** in **squared error loss**.
  - There is a probabilistic interpretation based on Normal Distributions.

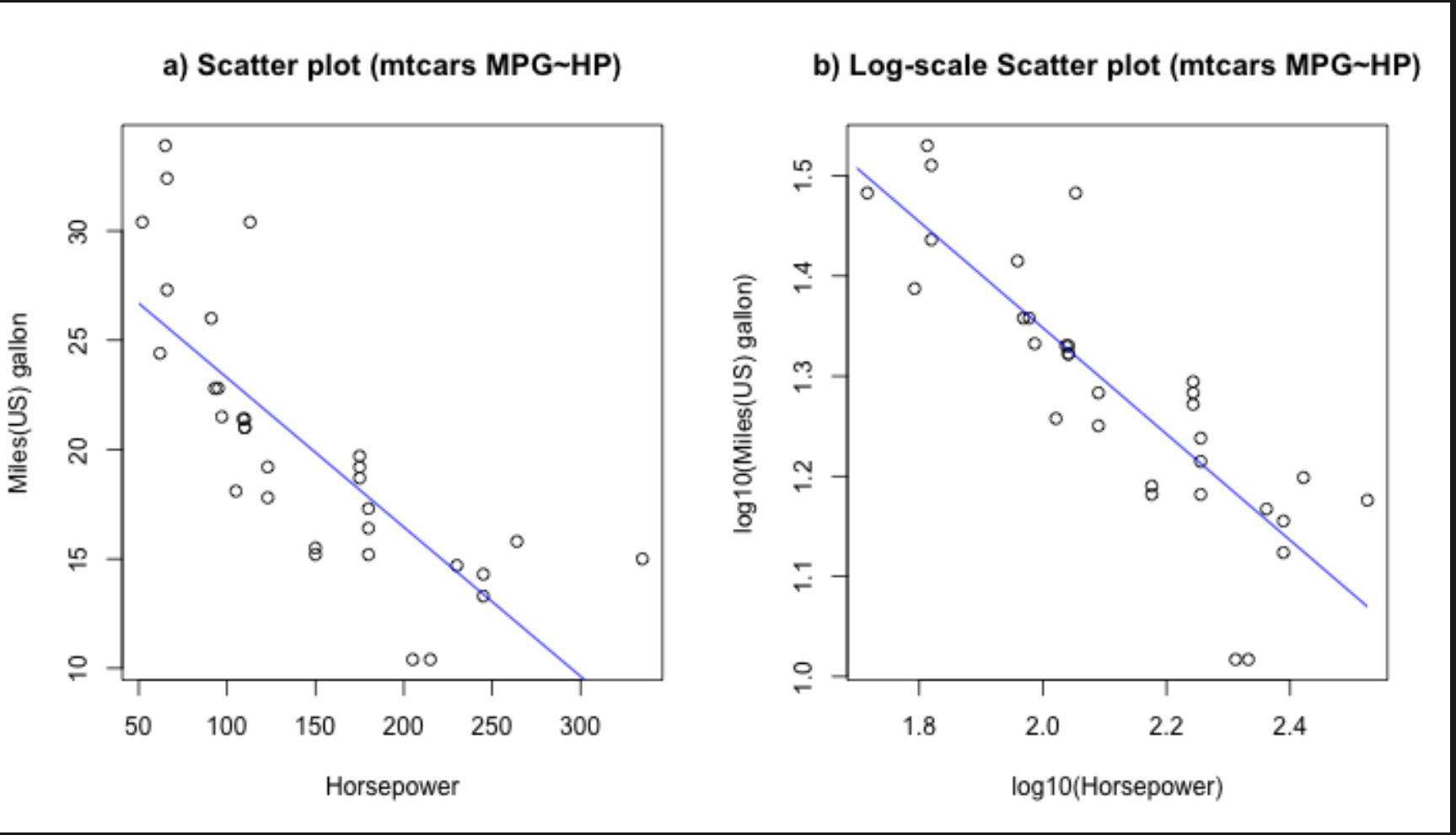
# (Not) Causality

- Regression is a often used as a tool to examine causality. . .
  - A and B share a causal relationship if a regression for B given A has an association, conditional on ("controlling for") C (C=**everything else**)
  - This does not resolve whether A causes B, or B causes A
  - Since we don't measure **everything else**, regression rarely establishes causality!
  - Further assumptions are needed to make a causal connection. This is known as **causal inference**.

# Discrete predictors

- If you include categorical/factor predictors, each **level** or unique value is used as a binary predictor.
- This is called **One Hot Encoding**.

# Regression example



# Multiple Regression example

```
> lm(mpg ~ cyl + hp + wt,data=mtcars) %>% summary

Call:
lm(formula = mpg ~ cyl + hp + wt, data = mtcars)

Residuals:
    Min      1Q  Median      3Q     Max
-3.9290 -1.5598 -0.5311  1.1850  5.8986

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 38.75179    1.78686  21.687  < 2e-16 ***
cyl         -0.94162    0.55092  -1.709 0.098480 .
hp          -0.01804    0.01188  -1.519 0.140015
wt          -3.16697    0.74058  -4.276 0.000199 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘
```

# Important measures of regression

- **R squared** (and adjusted R squared): variance explained/total variance. This tells us how predictable $y$ is.
- The coefficients $\beta_i$.
  - These should be compared to their error $\hat{\sigma}_i$.
  - The ratio is a t-value ($t_i = \beta_i / \hat{\sigma}_i$) from which a p-value can be calculated.
- F statistic and F test p-value:
  - $F$ is the ratio of the explained to unexplained variance, accounting for the **degrees of freedom**.
  - The full model compared to a null in which there are no explanatory variables.
  - Used in variable selection, ANOVA, etc.

# Regression is analogous to linear algebra with noise

- Most problems in Linear Algebra can be seen as **solving a system of linear equations:**

$$\mathrm{X}\mathrm{A} + \mathbf{b} = 0.$$

  - Where $\mathrm{X}$ is an $n$ by $p$ matrix of data,
  - $\mathrm{A}$ is an $p$ by $1$ matrix of coefficients,
  - and $-\mathbf{b}$ is a $n$-vector of target values.
- However, in the presence of noise we seek the least-bad fit:

$$\mathrm{argmin}_{(\mathrm{A},\mathbf{b})}||\mathrm{X}\mathrm{A} + \mathbf{b}||_2^2 = \sum_{i=1}^{N}(\mathbf{x}_i\mathrm{A} + b_i)^2$$

  - i.e. we find $\mathrm{A}$ and $\mathbf{b}$ such that they minimise the distance (in the squared $L_2$ norm)
- Linear algebra solves this very effectively!

# Matrix form of least squares

- Consider data $\mathrm{X}'$ with $p'$ **features** (columns) and $n$ observations.
- Given the regression problem:

$$\mathbf{y} = \mathrm{X}'\beta' + \mathbf{b} + \mathbf{e}$$

- to find:
  - $\beta'$ (a matrix dimension $p' \times 1$)
  - and $b$,
  - to minimise 'error': in $e^2 = \sum_{i=1}^{n} \epsilon_i^2$

# Matrix form of least squares

- We construct a simpler representation by adding a constant feature:

$$\mathrm{X} = \begin{bmatrix} 1 & \mathrm{X}_{11} & \cdots & \mathrm{X}_{1p'} \\ & & \cdots & \\ 1 & \mathrm{X}_{n1} & \cdots & \mathrm{X}_{np'} \end{bmatrix}$$

  - which has $p = p' + 1$ features.
- We now solve the analogous equation:

$$\mathbf{y} = \mathrm{X}\beta + \mathbf{e}$$

  - which has the same solution but is in a more convenient form.

# Mean Squared Error (MSE)

- The **prediction error** is:

$$\mathbf{e}(\beta) = \mathbf{y} - \mathrm{X}\beta$$

- Using the notation that $\mathbf{e}$ is a $p$ by $1$ matrix
- The **estimation error** is written in matrix form:

$$\mathrm{MSE}(\beta) = \frac{1}{n}\mathbf{e}^T\mathbf{e}$$

- Why? $\mathbf{e}^T\mathbf{e} = \sum_{i=1}^{n} e_i^2$
  - Hence $\mathrm{MSE}(\beta)$ is a $1 \times 1$ matrix, i.e. a scalar, and $|\mathrm{MSE}(\beta)| = \mathrm{MSE}(\beta)$.
  - Noticing this sort of thing makes the matrix algebra easier.
- We want to minimise this MSE with respect to the parameters $\beta$.

# How to do the Matrix Algebra

Lecture 13 of Cosma Shalizi's notes is a really helpful reminder!

- Look at the Matrix Algebra Cheat Sheet - specifically:
  - How does a transpose work?
  - How do you re-order elements?
  - How does a gradient work in linear and quadratic forms?

# Minimising MSE

- Taking (vector) derivatives with respect to $\beta$:

$$\nabla \mathrm{MSE}(\beta) = \frac{1}{n}(\nabla \mathbf{y}^T\mathbf{y} - 2\nabla\beta^T\mathrm{X}^T\mathbf{y} + \nabla\beta^T\mathrm{X}^T\mathrm{X}\beta) \quad (1)$$

$$= \frac{1}{n}(0 - 2\mathrm{X}^T\mathbf{y} + 2\mathrm{X}^T\mathrm{X}\beta) \quad (2)$$

- which is zero at the optimum $\hat{\beta}$:

$$\mathrm{X}^T\mathrm{X}\hat{\beta} - \mathrm{X}^T\mathbf{y} = 0$$

- with the solution:

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

- Exercise: For the case $p' = 1$, check that this solution is the same as you can find in regular linear algebra textbooks.

# Motivation: Residuals

- The **residual sum of squares** for $n$ predictions of a univariate $y$:

$$R^2 = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$$

- The expected value of the prediction error $\mathbb{E}(e^2) = R^2/n$.
- What happens if **compare two models** $M_1$ and $M_2$, where $M_1$ is a subset of $M_2$?

# Linear Models - Model selection

- For illustration, consider

$$Y = \mathbf{x}_1 A_1 + \epsilon_1$$

- and

$$Y = \mathbf{x}_1 A_1 + \mathbf{x}_2 A_2 + \epsilon_2.$$

- Unless $\mathbf{x}_2 = 0$ or $\mathbf{x}_2 \equiv \mathbf{x}_1$, then $\epsilon_2^2$ will be smaller than $\epsilon_1^2$.
  - This is an example of a more general rule: **larger models always have better predictions**.
- So prediction error is OK to use to fit models with the same dimension, but is incomplete for **model selection**.

# Cross-Validation Motivation

- Usually we are not interested in properties of **our sample**.
- We instead wish to know how our inference will generalise to **new samples**.
- The most straight forward way to predict how a model generalises is to test in **held-out data**.
- **Cross Validation** is a procedure to leave-out some data for testing.
- How much data?
  - **Leave-one-out Cross-Validation** (LOOCV) leaves out one datapoint at a time for testing.
  - **k-Fold Cross Validation** (k-fold CV) keeps a fraction $(k-1)/k$ of the data for learning parameters and $1/k$ for testing.

# Prediction accuracy in linear regression

- In linear regression, the errors are

$$\mathbf{e} = \mathbf{y} - \mathrm{X}\beta = \mathbf{y} - \hat{\mathbf{y}}$$

- We show in Worksheet 2.2A that the expected MSE for the $i$-th datapoint is:

$$\mathbb{E}(e_i^2) = \mathbb{E}\left[(y_i - \hat{y}_i)^T (y_i - \hat{y}_i)\right] = \mathbb{E}\left[(y_i - \hat{y}_i)^2\right] \tag{3}$$

$$= \mathrm{Var}[y_i] + \mathrm{Var}[\hat{y}_i] - 2\mathrm{Cov}[y_i, \hat{y}_i] + [\mathbb{E}(y_i) - \mathbb{E}(\hat{y}_i)]^2 \tag{4}$$

- This is shown by rearranging the formula for $\mathrm{Var}[y_i - \hat{y}_i]$

# Out-of-sample prediction accuracy in linear regression

- We can write the same thing when predicting an **out-of-sample** $y'_i$:

$$\mathbb{E}({e'}_i^2) = \mathbb{E}\left[(y'_i - \hat{y}_i)^T (y'_i - \hat{y}_i)\right] \tag{5}$$

$$= \mathrm{Var}[y'_i] + \mathrm{Var}[\hat{y}_i] - 2\mathrm{Cov}[y'_i, \hat{y}_i] + [\mathbb{E}(y'_i) - \mathbb{E}(\hat{y}_i)]^2 \tag{6}$$

- But out-of-sample, $\mathrm{Cov}[y'_i, \hat{y}_i] = 0$ whereas within-sample, $\mathrm{Cov}[y_i, \hat{y}_i] \neq 0$.
- Therefore:

$$\mathbb{E}({e'}_i^2) = \mathbb{E}(e_i^2) + 2\mathrm{Cov}[y_i, \hat{y}_i]$$

# Quantifying Out-of-sample prediction accuracy

- The mean out-of-sample prediction error can be rewritten (see Appendix) as:

$$\mathbb{E}(e'^2) = n^{-1}\sum_{i=1}^{n} e_i'^2 = n^{-1}\sum_{i=1}^{n} e_i^2 + 2n^{-1}\sigma^2 p$$

- The **optimism** is defined as $2n^{-1}\sigma^2 p$.
- The optimism grows with $\sigma^2$ and $p$ but shrinks with $n$. It is used to define the **model selection criteria** $\Delta C_p$ which is minimised:

$$\Delta C_p = MSE_1 - MSE_2 + \frac{2}{n}\hat{\sigma}^2(p_1 - p_2)$$

# Linear model optimism and AIC

- Minimising **Akaike's Information Criterion**:

$$AIC = -2\mathbb{L}(\hat{\theta}) + 2\mathrm{Dim}(\theta)$$

- reduces to maximising $\Delta C_p$ when the Likelihood $\mathbb{L}$ is a Normal distribution.
- There are many other Information Criteria...

# LOOCV

- We write a statistic $\hat{s}$ based on all data $\{y\}$ except $i$ as $\hat{s}^{(-i)}$ and the data is $\{y\}^{(-i)}$.
- For a general **loss function** we can write:

$$LOOCV = \frac{1}{n}\sum_{i=1}^{n} \text{Loss}\left(y_i; \hat{\theta}|y^{(-i)}\right)$$

- i.e. we evaluate the loss function for each datapoint using the estimate from the remaining data.
- NB **A loss function** is something that we choose the parameters $\theta$ to minimise. It can be:
  - the MSE,
  - the (negative log) likelihood,
  - a penalised version of these,
  - or any other convenient quantity.

# LOOCV for linear models

- For the MSE of a linear model we can write:

$$LOOCV = \frac{1}{n}\sum_{i=1}^{n}\left(y_i - \hat{y}_i^{(-i)}\right)^2$$

- It is not particularly straightforward[1] to show that:

$$LOOCV = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{y_i - \hat{y}_i}{1 - H_{ii}}\right)^2$$

- Where $\mathrm{H}$ is a function of $\mathrm{X}$ only (see Appendix).
- This is a very important quantity, often called the Studentized residual
- i.e. the LOOCV can be directly computed from a regression containing all data:
  - "downweighting" low-leverage data
  - "upweighting" high-leverage (hard to predict) data.

[1] Our references avoid proving this, but do discuss the motivation. Proofs are available but beyond scope.

# Leave-one-out Cross-Validation

- Leaving out a single datapoint is going to be insufficient unless the **data are independent**.
- The real world is rarely completely independent.
- However, there is often a computationally convenient way to compute LOOCV, and it is still better than leaving nothing out. It converges to $C_p$ for large $n$.
- Analogous tricks work for:
  - **Linear models** including **Best Linear Unbiased Predictors** (BLUPs)
  - **Kernel methods**
  - **Nearest neighbour** methods
  - And others

# Asymptotics

- Here are some facts about the asymptotic behaviour of LOOCV:
  - As $n \to \infty$, the expected out-of-sample MSE of the model picked by LOO cross-validation is **close to that of the best model** considered.
  - As $n \to \infty$, if the true model is among those being compared, LOOCV tends to pick a **strictly larger model** than the truth.
- LOOCV is not the right tool for choosing the **right model**.
- It is however an excellent tool for choosing the model with the best out-of-sample **predictive power**.
- . . . when the test data come from the **same distribution as the training data**!

# Implications

- Matrix form is a massive simplification of complex algebra
- It is easy to check that e.g. dimensions make sense
- These vector calculations are repeated in many machine-learning methods
- The details change but the principle remains
- Linear-Algebra loss minimisation techniques are extremely important
- They often sit inside a wider argument, e.g. updated conditional on some other parameters

# Reflection

- By the end of the course, you should:
  - Be able to define **correlation** and **regression** in multivariate context
  - Be able to perform basic calculations using these concepts
  - Be able to extend intuition about their application.
  - Be able to follow the reasoning in a paper where things get complicated.
- Matrix algebra is worth reading up on!
  - Describe it for example in your assessments' reflection.

# Signposting

- Make sure to look at **02.1-Regression.R**
- The mathematics behind Modern Regression is analogous to the mathematics underpinning scalable Machine Learning. **It is very important**.
- For accessible material see Cosma Shalizi's Modern Regression Lectures (Lectures 13-14)
- Further reading in chapters 2.3 and 3.2 of The Elements of Statistical Learning: Data Mining, Inference, and Prediction (Friedman, Hastie and Tibshirani)
- Next up: 2.2 Statistical Testing

# Appendix: The Hat Matrix

- There is an important and **response independent** quantity hidden in the prediction:

$$\mathrm{H} = \mathrm{X}(\mathrm{X}^T\mathrm{X})^{-1}\mathrm{X}^T$$

- The fitted values are:

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

  - $\mathrm{H}$ is dimension $N \times N$
  - $\mathrm{H}$ "projects" $\mathbf{y}$ into the fitted value space $\hat{\mathbf{y}}$
  - Put the "hat" on $y$

# Appendix: Properties of the Hat Matrix

- **Influence**: $\frac{\partial \hat{y}_i}{\partial y_j} = \mathrm{H}_{ij}$. So $\mathrm{H}$ controls how much a change in one observation changes the estimates of each other point.
- **symmetry**: $\mathrm{H}^T = \mathrm{H}$. So influence is symmetric.
- **Idempotency**: $\mathrm{H}^2 = \mathrm{H}$. So the predicted value for any projected point is the predicted value itself.
- You should read up on these and other vector algebra properties.

# Appendix: Residuals and the Hat Matrix

- The residuals can be written:

$$\mathbf{e} = \mathbf{y} - \mathrm{H}\mathbf{y} = (\mathrm{I} - \mathrm{H})\mathbf{y}$$

- $\mathrm{I} - \mathrm{H}$ is also symmetric and idempotent, and can also be interpreted in terms of Influence.
- Because of this,

$$\mathrm{MSE}(\hat{\beta}) = \frac{1}{n}\mathbf{y}^T(1 - \mathrm{H})^T(1 - \mathrm{H})\mathbf{y} = \frac{1}{n}\mathbf{y}^T(1 - \mathrm{H})\mathbf{y}$$

# Appendix: Expectations

- If the data were generated by our model(!) then they are described by an RV $\mathbf{Y}$ (an $n$-vector):

$$\mathbf{Y}_i = \mathbf{x}_i\beta + \epsilon_i$$

- $\mathbf{x}_i$ is still a vector but *not* a Random Variable!
- $\epsilon$ is an $n \times 1$ matrix of RVs with mean $\mathbf{0}$ and covariance $\sigma_s^2\mathrm{I}$.
- From this it is straightforward to show that the **fitted values are unbiased**:

$$\mathbb{E}[\hat{\mathbf{y}}] = \mathbb{E}[\mathbf{H}\mathbf{Y}] = \mathbf{x}\beta$$

- using the properties of Expectations with the symmetry and idempotency of $\mathrm{H}$.

# Appendix: Covariance

- Similarly, it is straightforward to show that

$$\mathrm{Var}[\hat{\mathbf{y}}] = \sigma_s^2 \mathrm{H}$$

  using the properties of Variances with the symmetry and idempotency of $\mathrm{H}$.
- In other words, the covariance of the fitted values is determined entirely by the structure of the covariates, via the Hat matrix.

# Appendix: Quantifying Out-of-sample prediction accuracy

- For the second term in $E({e'}_i^2) = \mathbb{E}(e_i^2) + 2\text{Cov}[y_i, \hat{y}_i]$,
- We're now able to compute the covariance between $y_i$ and its prediction $\hat{y}_i$:

$$\text{Cov}[y_i, \hat{y}_i] = \sigma^2 \text{H}_{ii}$$

- The mean out-of-sample prediction error is

$$\mathbb{E}({e'}^2) = n^{-1} \sum_{i=1}^{n} {e'}_i^2 = n^{-1} \sum_{i=1}^{n} e_i^2 + 2n^{-1}\text{tr}(\text{H})$$

- We show in Worksheet 2.2A that $\text{tr}(\text{H}) = \sigma^2 p$ where $p$=number of predictors.
- The **optimism** is defined as $2n^{-1}\sigma^2 p$.
- The optimism grows with $\sigma^2$ and $p$ but shrinks with $n$. It is used to define the **model selection criteria** $\Delta C_p$ which is minimised:

$$\Delta C_p = MSE_1 - MSE_2 + \frac{2}{n}\hat{\sigma}^2(p_1 - p_2)$$