Lecture 4: Model Selection

1. Review of the bias-variance trade-off
2. Model selection
3. Validation and cross-validation
4. Analytical criteria
5. Application example

QBUS6810: Statistical Learning and Data Mining

Lecture 4: Model Selection

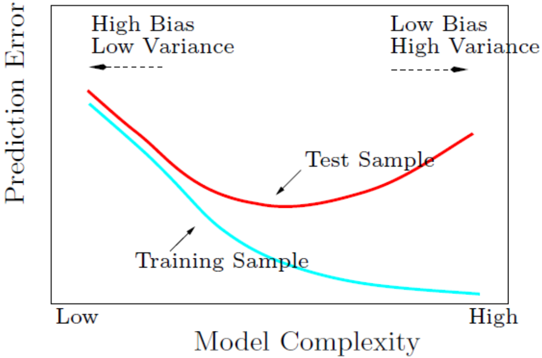
Semester 1, 2019

Discipline of Business Analytics, The University of Sydney Business School

Review of the bias-variance trade-off

Bias-variance trade-off

In Lecture 1 we discussed the fundamental concept of the bias-variance trade-off for estimation.



Bias-variance trade-off

- Increasing model complexity brings greater flexibility and, therefore, lower bias. However, this comes at a cost of higher variance. Overfitting can be a problem.

- Decreasing model complexity leads to lower variance. However, simpler models may not be sufficiently flexible to capture the underlying patterns in the data, leading to higher bias.

Examples

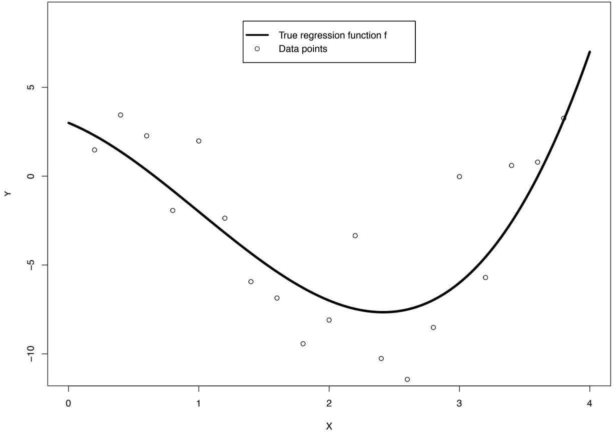
Linear regression. Adding predictors increases model complexity. Least squares estimates have greater variability when the number of predictors is large. On the other hand, excluding relevant predictors leads to bias.

KNN regression. Reducing the number of neighbours increases model complexity. Closer neighbours means lower bias. However, the fact that we average fewer observations leads to increased variance.

Simulated Example

The data is generated from the following (true) model:

$$Y = 3 - 3X - 3X^2 + X^3 + \varepsilon \qquad \text{where } \varepsilon \sim N(0,6)$$



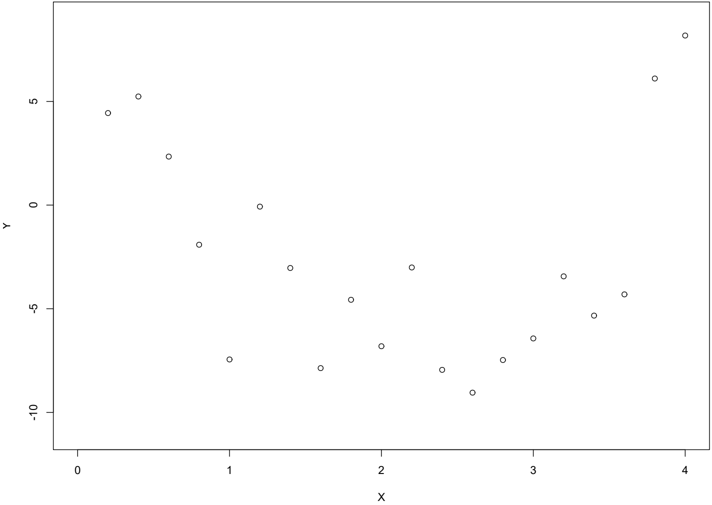
Simulated Example

The following three models are estimated using OLS:

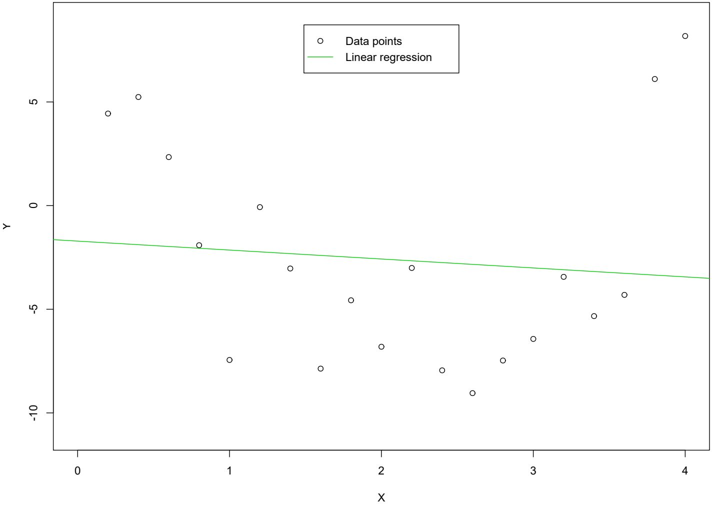
- Linear regression (X is the only predictor).
- Cubic regression (predictors: X, X^2 and X^3).
- 12-degree polynomial regression (predictors: X, X^2,...,X^{12}).

We will compare the estimated regression functions $\widehat{f}(x)$ to the true one: $f(x) = 3 - 3x - 3x^2 + x^3$.

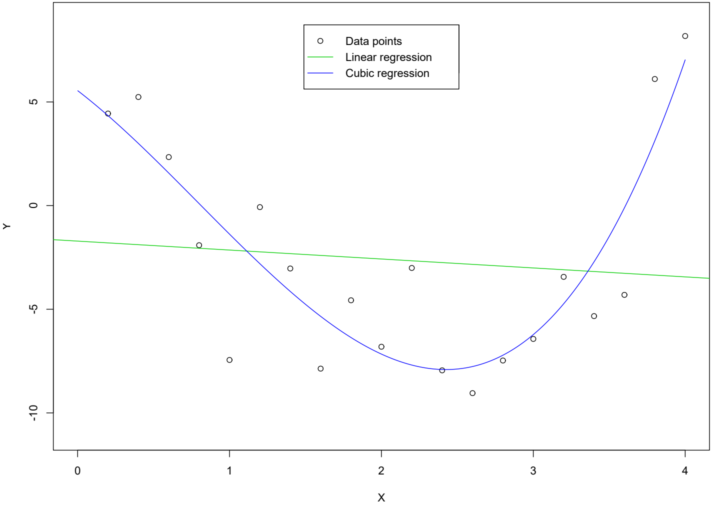
The Data



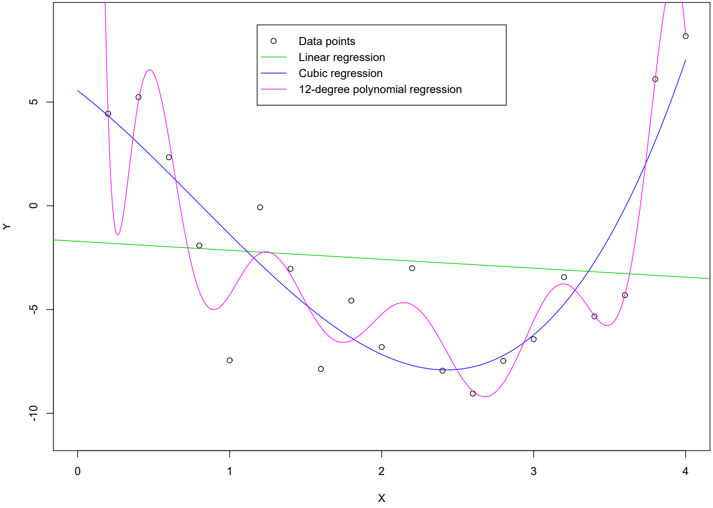
Linear fit



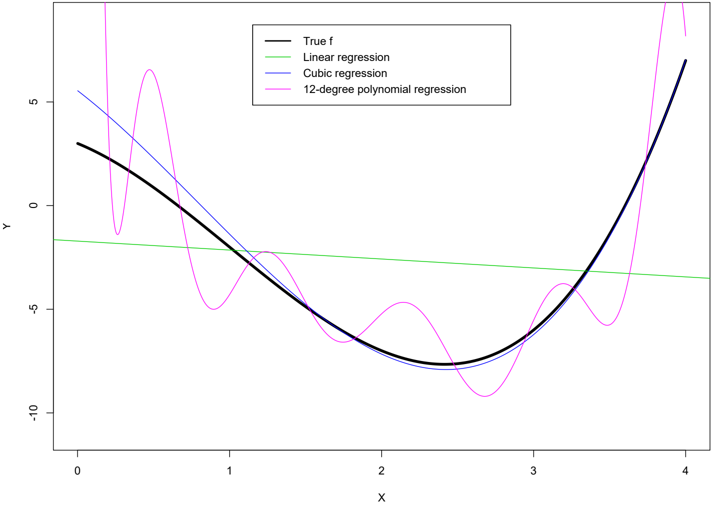
Linear and Cubic



All three fits to the data



Comparison with the true regression function f



Prediction error

We now focus on the prediction at $X = 2$.

The reducible part of the expected prediction error splits into squared bias and variance:

$$\begin{aligned}
\text{Reducible Error} &= E\left[(f(2) - \widehat{f}(2))^2\right] \\
&= \left(E[\widehat{f}(2)] - f(2)\right)^2 + E\left[\left(\widehat{f}(2) - E[\widehat{f}(2)]\right)^2\right] \\
&= \text{Bias}^2\left(\widehat{f}(2)\right) \quad + \quad \text{Var}\left(\widehat{f}(2)\right)
\end{aligned}$$

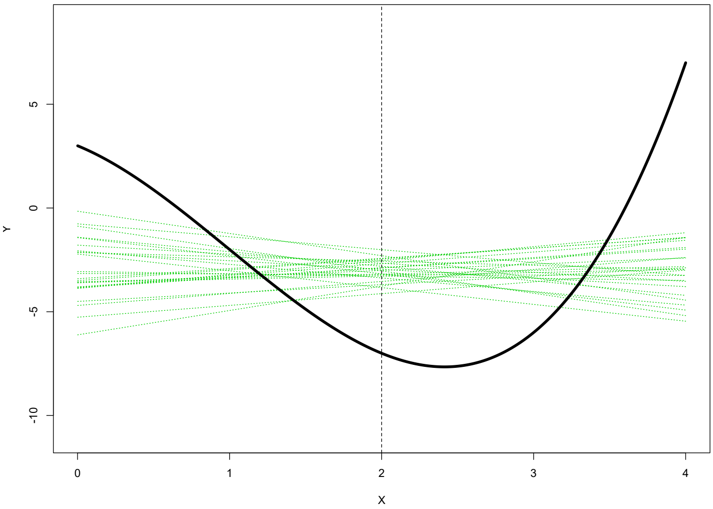
Prediction error

$$\text{Reducible Error} = \left(E[\widehat{f}(2)] - f(2)\right)^2 + E\left[\left(\widehat{f}(2) - E[\widehat{f}(2)]\right)^2\right]$$
$$= \text{Bias}^2\left(\widehat{f}(2)\right) \quad + \quad \text{Var}\left(\widehat{f}(2)\right)$$

We can approximate the Bias by generating many datasets from the true model and averaging the $\left[\widehat{f}(2) - f(2)\right]$ values produced for the different datasets.

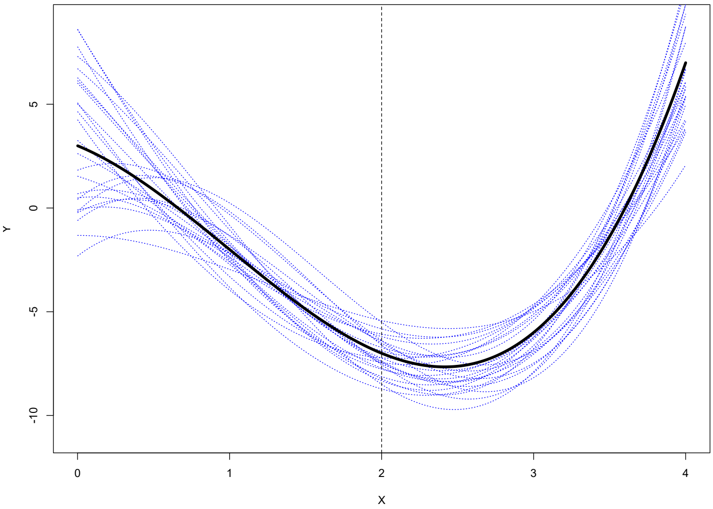
We can similarly approximate the variance by calculating the variance of $\widehat{f}(2)$ values produced for the different datasets.

Linear fit for 25 simulated datasets



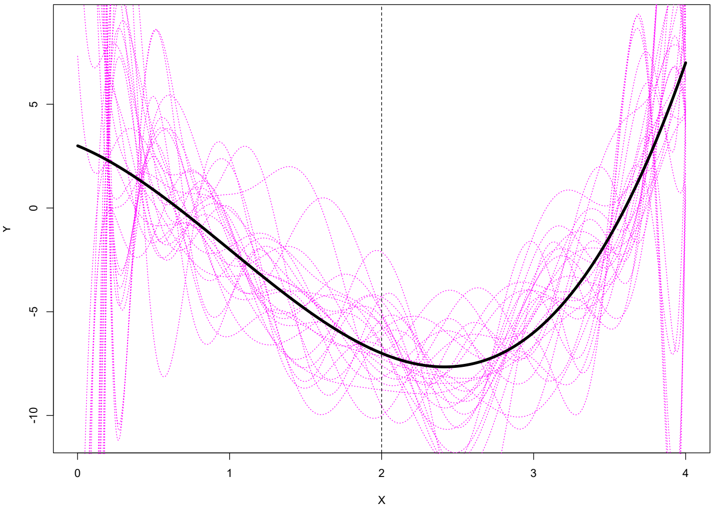
we see that $\widehat{f}(2)$ has high bias but low variance

Cubic fit for 25 simulated datasets



no more bias, but variance of $\widehat{f}(2)$ is somewhat higher

12-degree polynomial fit for 25 simulated datasets



still no bias, but the variance of $\widehat{f}(2)$ is now quite high

Prediction at $X = 2$

For illustration, we only repeated the estimation process 25 times, so the resulting approximations to the bias and variances are somewhat rough. Generating a larger number of datasets would improve the approximations.

	Linear	Cubic	12-degree
$\approx$ Bias^2	15.79	0.27	0.87
$\approx$ Variance	0.27	0.67	2.07
$\approx$ Reducible Error	16.06	0.94	2.94

Note that the actual (theoretical) bias in the cubic and 12-degree polynomial models is zero, because both models include all the true predictors and use OLS as the estimation method.

Model selection

Model selection

Model selection methods estimate the expected test error of a model based on the training data, allowing us to choose between models with different degrees of complexity.

We select the model that is estimated to have the best predictive ability.

Approaches to model selection

Validation set. Randomly splits the training set into two: one set for training the model, and another for testing the predictive performance and selecting the model complexity (this set is called a *validation* set).

Cross-validation. An efficient extension of the validation set approach that is based on multiple splits of the data into training and validation sets.

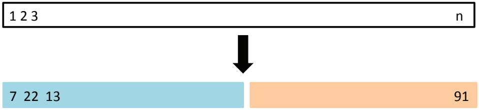
Analytical criteria. Uses analytical results to penalise the training error to account for overfitting.

Validation and cross-validation

Validation set

In the **validation set** approach, we randomly split the *training* data into two parts: another training set and a set we can use for testing the performance of the models (this is the *validation* set).

We select the model with the best predictive performance on the validation set.



Training, validation, and test split

Now, instead of just training and test data we split our data into 3 parts: training, validation and test. Here is how they are used:

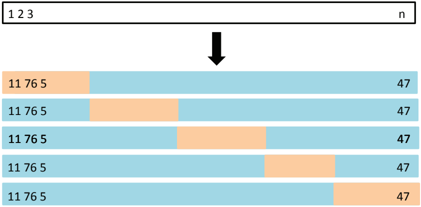
1. Estimate different models on the training data.
2. Make predictions on the validation set.
3. Select the model with the best validation set performance.
4. Re-estimate the selected model by combining the training and validation sets (i.e. going back to the original training data).
5. Assess the performance of the selected model by making predictions on the test data.

Cross-validation

The validation set approach has serious limitations when the size of the training data is not large. The model may not have enough data to train on, and there may not be enough cases in the validation set to reliably estimate the expected prediction error.

Cross-validation methods are based on multiple random training/validation splits. Unlike the validation set approach, each observation gets a turn at being predicted.

K-fold cross-validation



1. Randomly split the training sample into K **folds** of roughly equal size (note: $K = 5$ in the diagram above).
2. For each fold $k \in \{1, \ldots, K\}$, estimate the model on all other folds combined, then use the estimated model to make predictions and compute errors for all observations in fold k.
3. The cross-validation error is the average error across all the observations in the training sample.

K-fold cross-validation

5-fold and 10-fold CV. $K = 5$ or $K = 10$ folds are the most common choices for cross-validation.

Leave one out cross-validation. If we set $K = n$, this is called leave one out cross-validation, or **LOOCV**. For each observation i, we train the model on all other observations, and predict i.

Leave one out CV

Algorithm Leave one out CV for regression

```
for i=1:n do
    Assign observation $i$ to the validation set.
    Assign observations $1, \ldots, i-1, i+1 \ldots, n$ to the training set $\mathcal{D}_{-i}$.
    Estimate the model using the training set $\mathcal{D}_{-i}$.
    Use the model to compute the prediction $\widehat{f}^{-i}(\boldsymbol{x}_i)$.
    Compute the squared error $(y_i - \widehat{f}^{-i}(\boldsymbol{x}_i))^2$.
end for

Compute the leave-one-out MSE:
```

$$\text{MSE}_{\text{CV}_n} = \frac{1}{n} \sum_{i=1}^{n} (y_i - \widehat{f}^{-i}(\boldsymbol{x}_i))^2$$

LOOCV vs K-fold cross-validation

LOOCV. Approximately unbiased estimator of the expected prediction error. However, it can have high variance in some settings since the training sets are very similar for every prediction.

Furthermore, it can have a high computational cost as it requires us to fit the model n times (except in special cases).

K-fold. Lower computational cost and lower variance. It is subject to some bias because the training sets have sizes smaller than n.

Leave one out CV for linear regression

LOOCV estimate for the test error (reminder):

$$\mathsf{MSE}_{\mathsf{CV}_n} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \widehat{f}^{-i}(\boldsymbol{x}_i))^2$$

For the OLS method we can use a *shortcut* to compute the estimate above, without having to refit the model n times:

$$\mathsf{MSE}_{\mathsf{CV}_n} = \frac{1}{n}\sum_{i=1}^{n}\left[\frac{y_i - \widehat{f}(\boldsymbol{x}_i)}{1-h_i}\right]^2$$

where $\widehat{f}$ is the OLS estimate from the entire training set and h_i is the ith diagonal element of the matrix $\boldsymbol{H} = \boldsymbol{X}(\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T$

(so all h_i can be easily computed in one go).

Analytical criteria

Analytical criteria

Analytical criteria estimate the expected test error based on theoretical arguments. They have the form:

criterion = training error + penalty for number of parameters

We select a model that **minimizes** the value of the criterion.

Note that if we did't include a penalty and simply minimized the training error - the selected model would overfit.

Example: linear regression

It can be shown that in the case of linear regression the expected value of the training MSE underestimates the corresponding expected test MSE by

低估

$$\frac{\sigma^2}{n}2(p+1)$$

(we will use this fact without going into the proof)

Thus, we could make a correction to the training MSE by adding the above quantity to it. This would allow us to make a fairer comparison of different models.

Mallow's C_p statistic

The **Mallow's** C_p statistic applies to linear regression. It directly implements the recipe suggested on the previous slide (note that $MSE = RSS/n$).

We select the model with the lowest C_p, where

$$C_p = \frac{\text{RSS}}{n} + \frac{\widehat{\sigma}^2}{n} 2(p+1)$$

Here, $\widehat{\sigma}^2$ is an estimate of the variance of the error term in the regression model. This estimate is typically computed using the full model containing all the predictors.

通常使用包含所有预测变量的完整模型来计算该估计

Akaike Information Criterion

Mallow's C_p can be viewed as a special case of the **Akaike information criterion** (**AIC**), which is a popular and 多功能的 versatile strategy for model selection and applies to models estimated by maximum likelihood (this topic is discussed in the next lecture).

适用于通过最大似然估计的模型

In the case of linear regression with Gaussian errors:

$$\text{AIC} = \frac{1}{\widehat{\sigma}^2}\left(\frac{\text{RSS}}{n} + \frac{\widehat{\sigma}^2}{n}2(p+1)\right)$$

where $\widehat{\sigma}^2$ is again an estimate of the variance of the error term, typically computed using the full model with all the predictors.

Mallow's C_p and AIC

n 观察数 p 变量个数

Comparing

$$\text{AIC} = \frac{1}{\widehat{\sigma}^2}\left(\frac{\text{RSS}}{n} + \frac{\widehat{\sigma}^2}{n}2(p+1)\right)$$

and

$$C_p = \frac{\text{RSS}}{n} + \frac{\widehat{\sigma}^2}{n}2(p+1)$$

we see that AIC and C_p lead to the same selected model.

For practical purposes, the AIC and C_p are regarded as the same for linear regression.

Bayesian information criterion

iid独立同分布

The **Bayesian information criterion** (**BIC**) also applies to models estimated by maximum likelihood, and is derived from a *Bayesian* point of view (this approach is discussed in the next lecture).

In the special case of linear regression with Gaussian errors, BIC has a very similar form to AIC:

$$\text{BIC} = \frac{1}{\widehat{\sigma}^2}\left(\frac{\text{RSS}}{n} + \frac{\widehat{\sigma}^2}{n}\log(n)(p+1)\right)$$

Thus, BIC is proportional to AIC and C_p, but with a $\log(n)$ penalty factor instead of 2 (thus, typically a heavier penalty on complexity).

Model selection methods

LOOCV和AIC通常选择类似的模型，特别是当样本量很大时。
·AIC优于LOOCV的优势主要是计算。
·交叉验证是普遍适用的，而AIC则不是这样。
·当模型的假设（例如恒定误差方差）可能错误时，交叉验证应优先于AIC。

- LOOCV and AIC generally pick similar models, especially when the sample size is large.
- The advantage of AIC over LOOCV is mainly computational.
- Cross-validation is universally applicable, while this is not the case for AIC.
- Cross-validation should be preferred to AIC when the assumptions of the model (e.g. constant error variance) are likely to be wrong.

Limitations of model selection

- Standard statistical inference (e.g. confidence intervals, p-values) is no longer valid after model selection.
- This is because standard inference assumes a fixed model, whereas model selection will by definition pick a specific model that fits the data well.

Application example

Equity premium prediction data

Quarterly data from Goyal and Welch (2008), updated to 2015.

Response: quarterly S&P 500 returns minus treasury bill rate

Predictors (lagged by one quarter):

1. dp	Dividend to price ratio
2. dy	Dividend yield
3. ep	Earnings per share
4. bm	Book-to-market ratio
5. ntis	Net equity expansion
6. tbl	Treasury bill rate
7. ltr	Long term rate of return on US bods
8. tms	Term spread
9. dfy	Default yield spread
10.dfr	Default return spread
11.infl	Inflation
12.ik	Investment to capital ratio

Number of observations: 275 (1947-2015)

Complete subset regressions

- Suppose that we want to use linear regression to predict the equity premium. One option is to include all the $p = 12$ available predictors and estimate the model by OLS. However, the data are very noisy and this will lead to overfitting.

- The **complete subset regressions** (CSR) method is a simple and easy to understand algorithm that we can use to reduce overfitting.

- The CSR method fixes the model size k, then predicts the response by taking a simple average of the predictions produced by all possible linear regression models containing exactly k predictors. The number of such models is $\binom{p}{k}$, which is the number of all possible combinations of k out of p predictors.

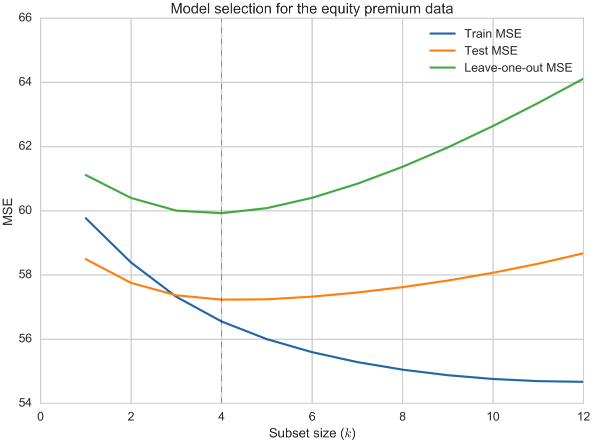
Complete subset regressions

Algorithm Complete subset regressions

```
Set $k$.
Generate all the $S = \binom{p}{k}$ possible predictor subsets of size $k$.
for *subset* in all subsets do
    Estimate the model by OLS based on the predictors included in *subset*. Denote the estimated regression function by $\widehat{f}_{\text{subset}}$ (note: the function only uses the values of the predictors in *subset*).
end for
The prediction for a new input vector $\boldsymbol{x}_0$ is
    $\widehat{y}_0 = (1/S) \sum_{\text{all subsets}} \widehat{f}_{\text{subset}}(\boldsymbol{x}_0)$
```

Equity premium prediction

We use leave-one-out cross validation to select the optimal model complexity (k) for the CSR method.



Equity premium prediction

- This data is characterised by a low signal-to-noise ratio. The evidence in the literature shows that the predictably of the equity premium is low.
- The optimal subset size according to leave-one-out cross validation is $k = 4$. This value of k also gives the lowest test MSE.
- Using all the inputs leads to poor predictions: the test R^2 is only 0.014 (compared to 0.04 for the CSR method).

Review questions

- How does model selection relate to the bias-variance trade-off?
- What is a validation set? How is it different from a test set?
- What is K-Fold cross validation? Describe how it works.
- What is the motivation for the C_p criterion and how does this criterion relate to AIC and BIC in the case of linear regression?