FRTN65 - Exam solutions 2023 Aug

1 Dimensional Analysis

a) We will apply Buckingham's Pi-theorem. To find dimensionless variables we need to find numbers a, b, c, d and e so that $\Pi = v^a \rho^b \mu^c \sigma^d D^e$ is unit-less. (Note that there are now two uses of the letter "d" in this solution, which might be unfortunate, but we will make sure that we do not mix these up). Introducing dimensions gives the requirement

$$[LT^{-1}]^a [ML^{-3}]^b [ML^{-1}T^{-1}]^c [MT^{-2}]^d [L]^e = M^0 L^0 T^0$$

which gives the equation system

$$\begin{matrix} L \\ M \\ T \end{matrix} \begin{bmatrix} 1 & -3 & -1 & 0 & 1 \\ 0 & 1 & 1 & 1 & 0 \\ -1 & 0 & -1 & -2 & 0 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \\ d \\ e \end{bmatrix} = 0.$$

The equation system has a 2-dimensional null space, spanned for example by the two vectors $(a, b, c, d, e) = (1, 1, -1, 0, 1)$ and $(a, b, c, d, e) = (2, 1, 0, -1, 1)$ (there are alternatives choices of these vectors). This corresponds to the variable combinations

$$\Pi_1 = \frac{\rho v D}{\mu} \quad \text{and} \quad \Pi_2 = \frac{\rho v^2 D}{\sigma}$$

Remark: Here Π_1 is called the Reynolds number, and Π_2 the Weber number.

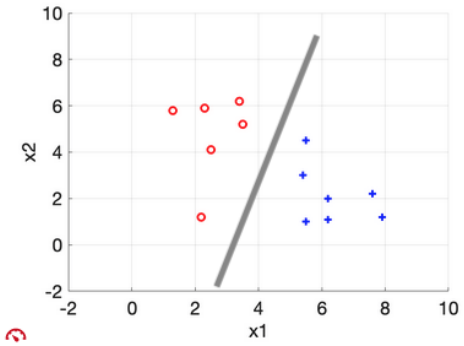
2 Logistic Regression

a) The function L is the log-likelihood of the data, so

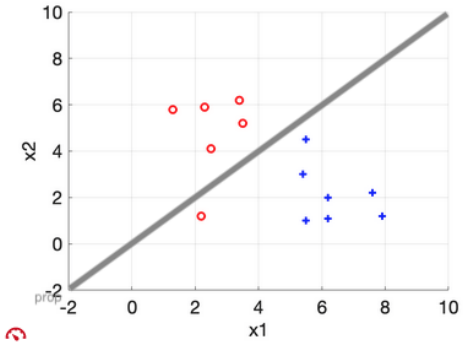
$$L(w, D_{train}) = \log \left(\sum_{\text{blue data}} \sigma(w_0 + w_1 x_1 + w_2 x_2) + \sum_{\text{red data}} (1 - \sigma(w_0 + w_1 x_1 + w_2 x_2)) \right)$$

Logistic regression finds the weight parameters w_0, w_1, w_2 that maximizes L, or equivalently minimizes $-L$. To classify future data points x, one can compare the value of $\sigma(w_0 + w_1 x_1 + w_2 x_2)$ with a threshold c. Here $c = 0$ is a natural choice (at least if both classes are considered equally likely apriorio, and if the cost of misclassification is the same for both classes).

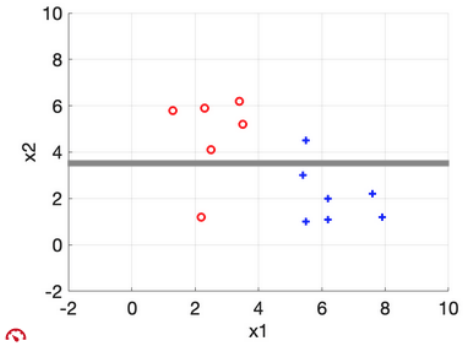
b) A typical decision boundary could look like this, resulting in 0 classification errors.



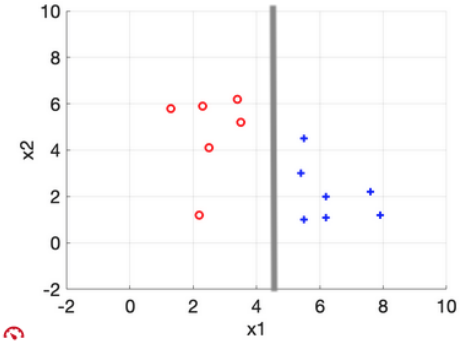
c) When w_0 is forced to be zero, the decision line will pass through the origin. A typical decision boundary would then result in result in 1 classification error, and look like the following



d) When w_1 is forced to be zero, the decision line can not depend on x_1 and therefore needs to be horizontal. The decision boundary would then result in 2 classification errors and look like the following



e) When w_2 is forced to be zero, the decision line needs to be vertical. The decision boundary can then result in result in 0 classification errors, and look like the following



3 Supervised Learning

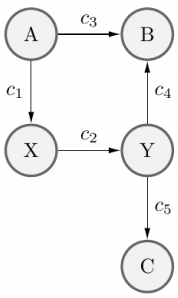
The most important observation is that the features have significantly different sizes. The feature in column 3 is roughly 1000 times larger than other features. This impacts the performance of the kernel ridge regression method, which penalizes large parameters and therefore makes the method prefer using mainly column 3 and disregard the other features. This size issue can be fixed by normalizing the input data (featurewise), for instance using this code

```
scaler = StandardScaler();
scaler.fit(Xtrain);
Xtrain = scaler.transform(Xtrain)
Xtest = scaler.transform(Xtest)
```

You can find [example of solution code here], achieving 84 percent accuracy with the KRR method. Also a random forest regressor is tested, achieving 85 percent accuracy. Also shown in this google colab file is the performance of a boosting method (just briefly mentioned in the course) which achieves almost 88 percent.

4 Causal Inference

a)



b) Will work. There is no open backdoor path (B is a collider)

c) Will work. There is no open backdoor path.

d) Not ok. The backdoor path is now opened.

e) Not ok. The backdoor criterion will not work since C is a descendent of X, which is not allowed. Also a thought experiment where the variable C is a noise free copy of Y (C=Y) will demonstrate that the OLS will result in $Y = 0 \cdot X + 1 \cdot C$ giving the false impression that the causal effect of X on Y is zero.

f) Not ok. Even though the backdoor path closed at A, threre is now a descendent of X included in the set (the variable B), so similar issue as in e).

g) There is not any major problem with including a bias in the estimation. The LS method will correctly identify that this bias is (close to) zero given enough data. The only minor drawback with estimating one more parameter than what is needed is that the estimation performance will be worse, since one estimates a not needed coefficient.

You can find [google colab code here] which verifies these conclusions.

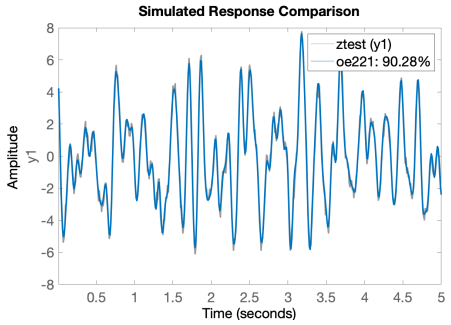
5 System Identification

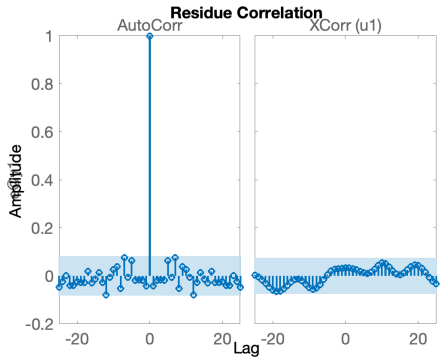
A visual inspection indicates five highly suspected outliers: At times 1.5,3.5,5.5,7.5 and 9.5 the y-value are all -13 and deviate much from adjacent values. It will improve performance to replace these values with interpolated values, for example taking the mean of adjacent values, or simply replace them with the previous y-value.

It is good practice to split the sequence into train and test data, for instance using a a 50/50 split. In this specific problem a specific test signal (a measured step response) was also available, which could alternatively be used to verify performance on independent data.

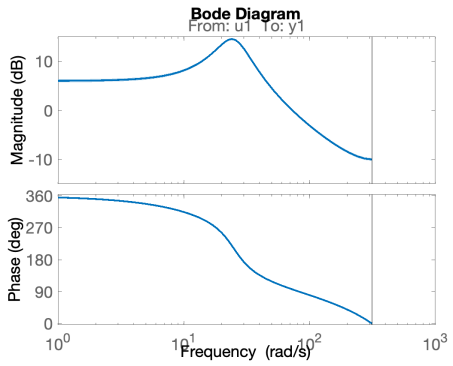
The ARX models all have rather bad accuracy, and an investigation of the residual plots indicate that the estimation error indicates residual covariance structure outside the acceptable confidence intervals.

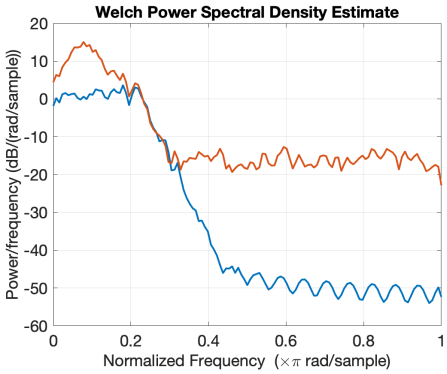
Some experiments with OE models quickly give that an oe221 structure give good results with 90 percent of the signal correctly explained by the model and reasonable residual plots.



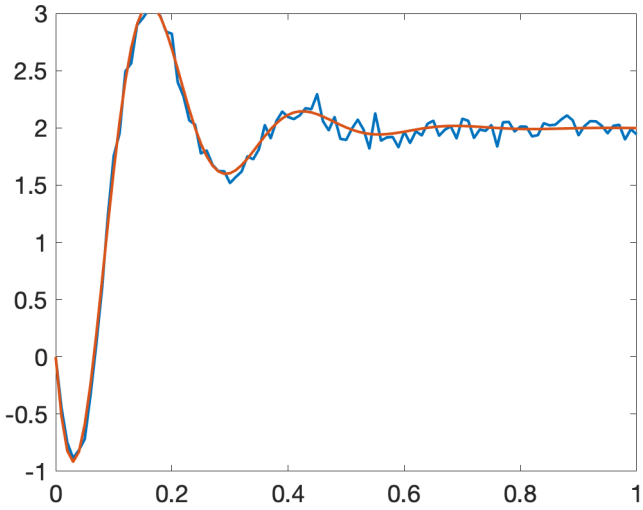


The Bode plot looks reasonable, and the spectral contents of input (blue) and output (red) indicate that there is good excitation up to 30 percent of the Nyqyuist frequency (i.e. 10-15 Hz, say 100 rad/s) so the identification of the resonance peak should be reliable.





A step response was also given, to be used as test data. The model output fits the data quite well, as seen in the figure below.



The identified parameters also seem to be estimated with good accuracy

```
oe221 =
Discrete-time OE model: y(t) = [B(z)/F(z)]u(t) + e(t)

  B(z) = -0.5197 (+/- 0.004882) z^-1 + 0.6355 (+/- 0.005026) z^-2

 F(z) = 1 - 1.802 (+/- 0.001266) z^-1 + 0.8598 (+/- 0.001233) z^-2
```

This turns out to be very close to the true system which was

```
sysd =

    -0.5152 z + 0.6306
  ----------------------
  z^2 - 1.803 z + 0.8607
```

6 System Identification Theory

a)

$$Y = \begin{bmatrix} y(1) \\ \vdots \\ y(N) \end{bmatrix} = \begin{bmatrix} \exp(u(1)) & \exp(-u(1)) \\ \vdots & \vdots \\ \exp(u(N)) & \exp(-u(N)) \end{bmatrix} \begin{bmatrix} \alpha \\ \beta \end{bmatrix} = X\theta$$

b)

Since the measurement equation is linear on the form $Y = X\theta + e$ where the noise e is independent Gaussian noise (e_k is independent both of other e_n and also of the elements in X) with mean zero it is a standard result from the course that the resulting estimation error is normally distributed as

$$\widehat{\theta}_N - \theta_0 \in N(0, R_N^{-1}), \quad \text{with } R_N = X^T X = \begin{bmatrix} \sum_{k=1}^N \exp(2u(k)) & \sum_{k=1}^N \exp(0)) \\ \sum_{k=1}^N \exp(0) & \sum_{k=1}^N \exp(-2u(k)) \end{bmatrix},$$

which can be rewritten as

$$\widehat{\theta}_N - \theta_0 \in N(0, \frac{1}{N}\bar{R}_N^{-1}), \text{with } \bar{R}_N = \frac{1}{N}X^T X = \begin{bmatrix} \frac{1}{N}\sum_{k=1}^N \exp(2u(k)) & 1 \\ 1 & \frac{1}{N}\sum_{k=1}^N \exp(-2u(k)) \end{bmatrix}.$$

When N becomes large we will, by the law of large numbers, have that

$$\bar{R}_N \to \begin{bmatrix} E\exp(2u) & 1 \\ 1 & E\exp(-2u) \end{bmatrix} = \begin{bmatrix} \exp(2) & 1 \\ 1 & \exp(2) \end{bmatrix},$$

where the last equality follows from the hint that $E(\exp(\gamma u)) = \exp(\gamma^2/2)$. To summarize, the estimation is bias-free for any N, and asymptotically one has

$$\sqrt{N}(\widehat{\theta}_N - \theta_0) \to N(0, \bar{R}^{-1}), \quad \text{when } N \to \infty.$$