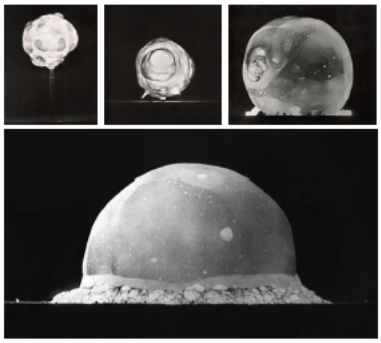
FRTN65 - Exam solutions 2022 Jan

1 Dimensional Analysis



Historical background: On July 16 1945, the first atomic bomb ever was detonated in New Mexico. These pictures were released and published in Life Magazine. The energy of the blast was however a highly classified secret. The British physicist Geoffrey Ingram Taylor, see `https://en.wikipedia.org/wiki/G._I._Taylor`, used dimensional analysis to estimate the energy from the data available in the pictures. This exam problem is about how it was done. For further description check out `https://www.youtube.com/watch?v=fEBOuN49xPs`

a) We will apply Buckingham's Pi-theorem. Dimensions of involved variables are

Quantity	**Unit**	**Dimension**
E	Joule	ML^2T^{-2}
R	Meter	L
t	Second	T
ρ	kg/m^3	ML^{-3}

To find dimensionless variables we need to find exponents a, b, c and d so

$$\begin{aligned}\Pi = E^a R^b t^c \rho^d = const \quad &\Leftrightarrow \quad (ML^2T^{-2})^a L^b T^c (M^L{-}3)^d = M^0L^0T^0 \\ &\Leftrightarrow \quad M^{(a+d)}L^{(2a+b-3d)}T^{(-2a+c)} = M^0L^0T^0.\end{aligned}$$

This can be written as a linear system of equations

$$\begin{array}{c} \\ M \\ L \\ T \end{array} \begin{array}{c} \begin{array}{cccc} E & R & T & \rho \end{array} \\ \begin{bmatrix} 1 & 0 & 0 & 1 \\ 2 & 1 & 0 & -3 \\ -2 & 0 & 1 & 0 \end{bmatrix} \end{array} \begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix} = 0$$

It is easy to see that the first three columns of the matrix are independent, and hence the rank is 3. There will therefore be a 4-3 = one-parametric null-space, so that one dimensionless Π-variable can

be formed. This null space can be found either by `null(A)` in matlab or by the calculation

$$\begin{cases} a &= -d = -t \\ b &= -2a + 3d = 5t \\ c &= 2a = -2t \\ d &= t. \end{cases}$$

With $t = 1$ we get a solution represented by integers $a = -1$, $b = 5$, $c = -2$ and $d = 1$, indicating the physical relation

$$\frac{R^5\rho}{Et^2} = \text{const.}$$

b.)

Plotting R vs t in a log-log diagram gives an almost straight line satisfying

$$\log R = k \log t + b$$

with slope $k \approx 0.4 = \frac{2}{5}$, which corresponds well with the obtained relation in a). Numerical least squares line-fitting could be done in matlab with the command

```
>> p = polyfit(log([3 5 15 60]),log([57 70 106 180]),1)
```

An alternative acceptable way is to check if R^5/t^2 is roughly the same for the data.

$t[s \cdot 10^{-3}]$	3	5	15	60
$R[m]$	57	70	106	180
$R^5/t^2[m^5/s^2]$	$6.7 \cdot 10^{13}$	$6.7 \cdot 10^{13}$	$5.9 \cdot 10^{13}$	$5.2 \cdot 10^{13}$

Since the values are roughly constant, considering the accuracy of the given data, this also supports the relation found in a).

c.)

$$\frac{R^5\rho}{Et^2} = 1 \implies E = \frac{R^5\rho}{t^2}.$$

With $\rho = 1\, kg/m^3$, the estimate of E can for example be roughly approximated as the mean of the R^5/t^2 values from calculated in b.). This mean is about $6 \cdot 10^{13}$ Joules (which was rather close to the true, secret, value for the bomb).

d.)

The dimension of pressure is now also needed, it is $[ML^{-1}T^{-2}]$.

$$R^f E^g \rho^h = P \implies L^f (ML^2T^{-2})^g (ML^{-3})^h = M^{(g+h)} L^{(f+2g-3h)} T^{(-2g-3h)} = ML^{-1}T^{-2}.$$

This gives the system of equations

$$\begin{cases} g+h & = 1 \\ f+2g-3h & = -1 \\ -2g-3h & = -2 \end{cases}$$

with solution

$$\begin{cases} f & = -3t \\ g & = t \\ h & = 0 \end{cases}$$

This means that the pressure does not depend on the density, and we get the relation

$$P = \frac{E}{R^3}.$$

Remark: Knowing that buildings can not resist a shock wave with magnitue around 1 bar = 10^5 Pa this could be used to predict destruction within a radius of about 1 km. In the case of the Hiroshima bombing, with similar energy, the reported radius of complete destruction was about one English mile.

2 Supervised Learning

The data was from the recent research article "Machine Learning can predict survival of patients with heart failure from serum creatine and ejection fraction alone" [Chicco and Jurman 2020], see this link.

a) The model is evaluated on the same data as it is trained, which is bad. The performance estimate is therefore highly optimistic, and wrong conclusions on good methods and hyperparameters will be made. The kNN models become overtrained.

b) Example on code investigating this data and the performance of different ML methods can be found for example here, or here.

A good solution should consist of some reasonable investigation of the data and its features and splitting the data into reasonably sized separate train and validations sets, or using cross-validation. For some methods one should preferably also scale features. Reasonable choices of hyperparameters should be done and discussed. The data set is unbalanced (about 68 percent survive, and 32 percent die), and the impact of making errors might be different for false positives and false negatives, therefore one should preferably evaluate results using not only accuracy but also with e.g. a confusion matrix, or AUC score or ROC curve. (The trival classifier "all will survive" gets an accuracy of 68 percent, but would fail to identify the fatilities, which might be of more importance to find). Typical reported accuracy is in the order of 80-85 percent.

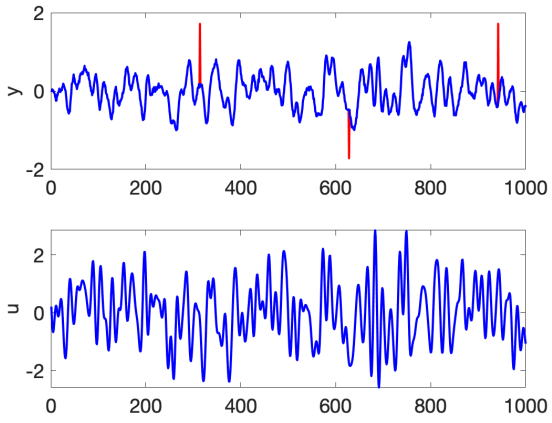
c) The original article reports an ROC AUC value of 0.79 for a gradient boosting method using only the two features "se1" (which stands for serum creatine a waste product generated when a muscle breaks down. A high value may indicate kidney problems) and "ej" (ejection fraction, the

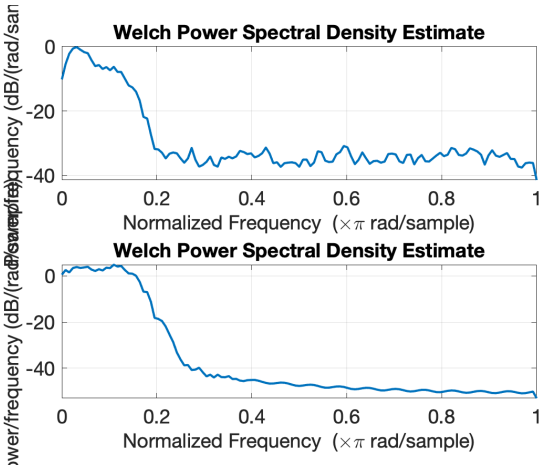
percentage of how much blood the left ventricle pumps out with each contraction, a low value is usually bad). Other choices are also possible, answers will be judged by if a reasonable procedure for the choice was done and motivated.

3 System Identification Hands-on

Plotting u and y we easily spot three outliers at time 314, 2*314, and 3*314. The corresponding three y-values all have absolute value 1.718281828. This can not be the true y-values and must come from some sensor error (or a malevolent teacher trying to complicate life for students...). These three y-values should preferably be replaced by e.g. the mean value of the adjacent samples. If the outliers are not spotted early in this initial phase, they should at least be easily spotted later in the residual analysis, when the model errors are plotted and studied. They are more than 10 standard deviations larger than the residual noise.

It is also a good idea to check excitation by plotting the spectra of input(lower plot) and outputs(upper plot). We see that we only have excitation up to about 0.15 of the Nyquist frequency, i.e. $0.15\pi/0.1 \approx 5$ rad/s.





Let us split the data into training and test sets

```
N = length(y);
ytrain = y(1:N/2);
utrain = u(1:N/2);
ytest = y(N/2+1:end);
utest = u(N/2+1:end);
ztrain = iddata(ytrain,utrain,h);
ztest = iddata(ytest,utest,h);
```

To get an initial idea of model structure one can start by fitting ARX models

$$A(z)y = B(z)u + e$$

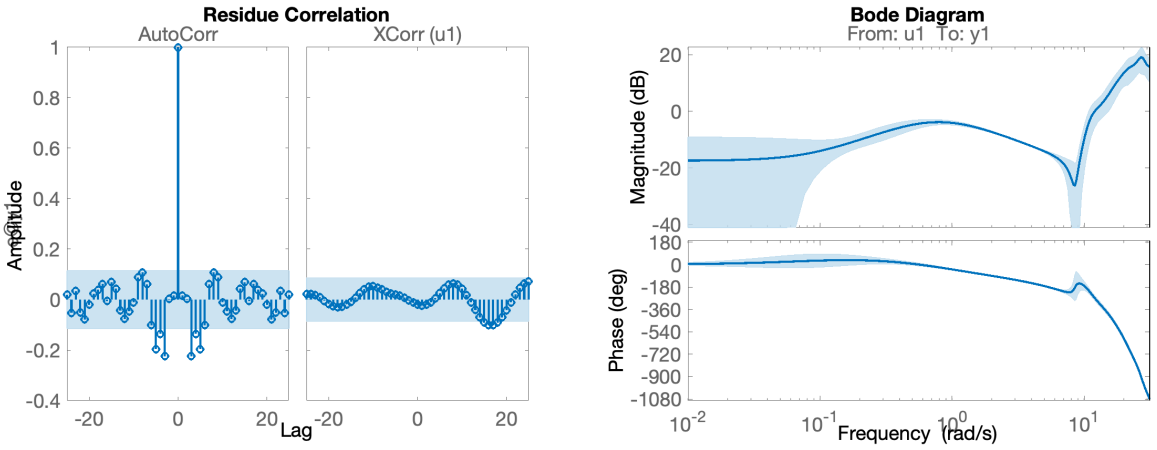
of different orders. This is most easily done by the `arxstruc` and `selstruc` command.

```
NN1 = struc(1:10,1:10,1:10);  % tries 1000 different models
V = arxstruc(ztrain,ztest,NN1);
Nbest = selstruc(V)
arxbest = arx(ztrain,Nbest)
```

To investigate the performace of the suggested best ARX model, a 10th order model, we can run

```
Nbest = [10 4 6];
arxbest = arx(ztrain,Nbest)
present(arxbest)
compare(ztest,arxbest,Inf)'
fig4 = bodeplot(arxbest,{0.01,pi/h})
showConfidence(fig4,3)
resid(ztest,arxbest)
```

The fit to data turns out to be 90 percent which is nice, but residual plots indicate a signficant remaining model error in both the autocorrelation of errors, and the correlation between error and input signal.



Furthermore, the Bode diagram of this ARX system has a very large amplitude peak for high frequencies, where we lack reliable excitation and hence we should be suspicious about the identification results. Note that the shaded confidence regions in the Bode diagram are calculated under the assumption that the model structure is correct, so they can not be trusted in case the model structure is wrong. In conclusion we decide that the ARX model structure does not work sufficiently well and should not be trusted.

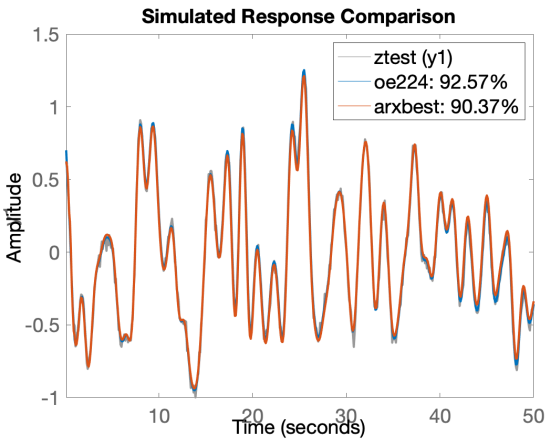
After instead trying the output error (OE) structure

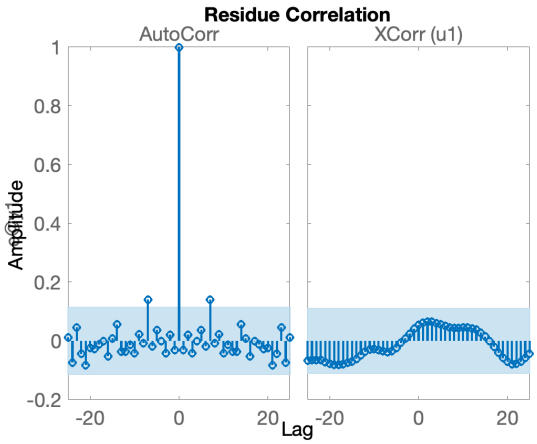
$$y(t) = \frac{B(z)}{F(z)}u(t - nk) + e(t)$$

we find after some trial and error a better fit for a 2nd order model, with a delay of nk=4. This good model needs only 4 parameters. (Trying the more general BJ or ARMAX structures gave no improvements). Note that using nk=1 would lead to models with good data fit but with unnecessary many parameters, typically in both B and F. The issue can often be seen by large standard deviations in the coefficients of B or F. This can sometimes give strange, unreliable, behavior in the identified Bode diagram for frequencies above 5 rad/s where the excitation is poor.

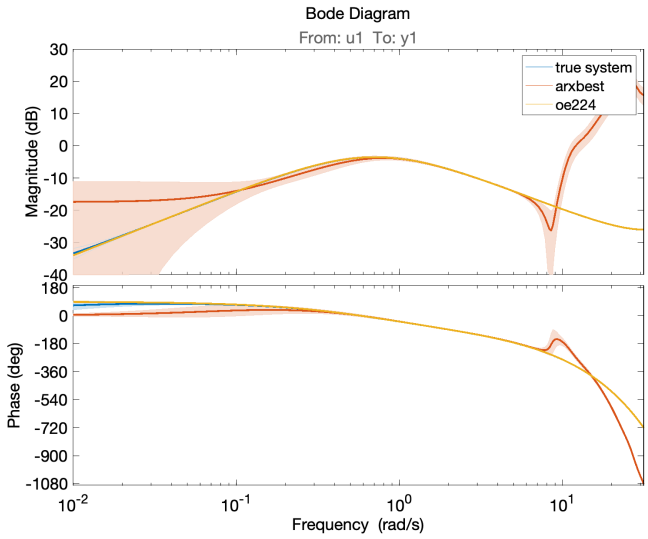
```
N = [2 2 4];
oe224 = oe(ztrain,N);
compare(ztest,oe224,arxbestInf)
fig = bodeplot(oe224,{0.01,pi/h})
showConfidence(fig,3)
resid(ztest,oe224)
```

This results in some percent better model fit and, more importantly, residuals that are statistically acceptable (the shaded regions correspond to 2 standard deviations, so 1-2 samples outside the region is what can be expected from random chance).





The following figure shows the Bode diagrams of the actual (unknown) true system (blue), oe224 (yellow) and arxbest (red). The blue and yellow curves are on top of each other, indicating a good model fit. The ARX model however has significant model errors at higher frequencies where the input excitation is poor and the wrong model structure has resulted in a significant bias due to influence of the noise.



The final suggested model is therefore

```
present(oe224)
oe224 =
Discrete-time OE model: y(t) = [B(z)/F(z)]u(t) + e(t)

  B(z) = 0.09279 (+/- 0.0004878) z^-4 - 0.09275 (+/- 0.0004888) z^-5

  F(z) = 1 - 1.856 (+/- 0.000962) z^-1 + 0.8608 ( +/- 0.0009624) z^-2
```

It is a good sign that all identified coefficients in B and F have very low standard deviations compared to their identified values.

Remark: The true system was given by the following 2nd order system with a time delay. There was 8 percent white noise added on the output.

```
sysc = 2*s/(1+s)/(1+2*s)*exp(-0.3*s);
sysd = c2d(sysc,h)
sysd =

            0.09278 z - 0.09278
  z^(-3) * ----------------------
           z^2 - 1.856 z + 0.8607
```

This is very close to our identified OE model

4 System Identification Theory

a) The parameters occur linearly and the model can be written

$$y = \theta^T \phi(t) + e(t), \quad i = 1, \ldots, N$$

where $\theta = [a \quad b]^T$, $\phi(t) = [u(t) \quad \exp(u(t))]^T$ and $e(t) \in N(0,1)$.

The standard least squares linear regression estimate is $\hat{\theta}_N = R_N^{-1} f_N$, where

$$R_N = \frac{1}{N}\sum_{t=1}^{N} \phi(t)\phi^T(t) \qquad \text{and} \qquad f_n = \frac{1}{N}\sum_{t=1}^{N} \phi(t)y(t).$$

The signal $y(t)$ is measured directly and $\phi(t)$ is obtained from the measured $u(t)$.

b) Since $u(t)$ and $e(t)$ are independent therefore also $\phi(t)$ and $e(t)$ are independent and also uncorrelated. Because of that, the estimate $\hat{\theta}_N$ is unbiased and

$$\sqrt{N}\left(\hat{\theta}_N - \theta_0\right) \in \text{As}N(0, \bar{R}^{-1})$$

where $\bar{R}$ is what R_N converges to when $N \to \infty$ and AsN is an asymptotic normal distribution, meaning that $\hat{\theta}_N$ is asymptotically Gaussian with mean value θ_0 and covariance matrix which for large N is close to $\frac{1}{N}\bar{R}^{-1}$).

$\bar{R}$ can be calculated with the law of large numbers

$$\begin{aligned}
\bar{R} &= \lim_{N\to\infty} R_N = E\left[\phi(t)\phi^T(t)\right] \\
&= E\left[\begin{bmatrix} u(t) \\ exp\,(u(t)) \end{bmatrix}\begin{bmatrix} u(t) & \exp(u(t)) \end{bmatrix}\right] = \\
&= \begin{bmatrix} E\left[u(t)^2\right] & E\left[u(t)\exp(u(t))\right] \\ E\left[u(t)\exp(u(t))\right] & E\left[\exp(2u(t))\right] \end{bmatrix}
\end{aligned}$$

To calculate the expectation we use that u is uniformly distributed between 0 and 1

$$\begin{aligned}
E\left[u^2\right] &= \int_{-\infty}^{\infty} u^2 p_u(u)du = \int_0^1 u^2 du = \frac{1}{3} \\
E\left[u\exp(u)\right] &= \int_0^1 u\exp(u)du = 1 \\
E\left[\exp(2u)\right] &= \int_0^1 \exp(2u)du = \frac{1}{2}\left(e^2 - 1\right) \\
\implies \bar{R} &= \begin{bmatrix} \frac{1}{3} & 1 \\ 1 & \frac{1}{2}\left(e^2-1\right) \end{bmatrix}
\end{aligned}$$

(In matlab, integrals can be calculated symbolically using e.g. `syms u; int(u*exp(u),0,1)`)

This means that the error covariance matrix for large N is close to

$$\frac{1}{N}\bar{R}^{-1} = \frac{1}{N}\frac{1}{\frac{1}{6}\left(e^2-1\right) - 1}\begin{bmatrix} \frac{1}{2}\left(e^2-1\right) & -1 \\ -1 & \frac{1}{3} \end{bmatrix} \approx \frac{1}{N}\begin{bmatrix} 49 & -15 \\ -15 & 5 \end{bmatrix}$$

We notice that the error variance of the b parameter is close to $5/N$, which is smaller than that for the a parameter which is roughly $49/N$. The correlation between the errors of the parameters is negative $-15/N$.

c) Yes it is the ML estimate. This is a standard result which only relies on that the parameters θ occur linearly, and that the noise e is additive, Gaussian, and independent of the regressors ϕ.

5 Causal Inference and DAGs

a) The correct expression is $y \sim x1 + x2 + x3$. When estimating the casual effect from one variable, $x3$, to another, y, it is important that there is no open backdoor path from $x3$ to y. All such backdoor paths must be blocked. In the expressions $y \sim x3$ and $y \sim x2 + x3$ backdoor paths are open and the estimates become wrong. This is also easily verified numerical by the code.

When doing linear regression to estimate the causal effect of $x3$ on y, the expression needs to include the variables of a valid adjustment set. One such set is the parents of $x3$, which are $x1$ and $x2$. This is why the expression $y \sim x1 + x2 + x3$ works. The other expressions will not work.

b) Since $x1$ has no parents the expression $y \sim x1$ can be used. From the linear structure we get

$$\begin{aligned} y &= c41 \cdot x1 + c42 \cdot x2 + c43 \cdot x3 + bias + ny = \\ &= c41 \cdot x1 + c42\,(c21 \cdot x1 + n2) + c43\,(c31 \cdot x1 + c32\,(c21 \cdot x1 + n2) + n3) + bias + ny = \\ &= (c41 + c42 \cdot c21 + c43 \cdot c31 + c43 \cdot c32 \cdot c21)x1 + \\ &\quad + bias + ny + c42 \cdot n2 + c43 \cdot n2 + c43 \cdot c32 \cdot n2 \end{aligned}$$

With the coefficient values as given in the google colab code, $c21, c31, c32, c41, c42) = (0.5, 0.3, 1.0, 0.8, 0.6)$, we get that the causal effect from x1 is

$$\frac{\partial}{\partial x} E[y \mid \mathbf{do}(x_1 := x)] = c41 + c42 \cdot c21 + c43 \cdot c31 + c43 \cdot c32 \cdot c21 = 1.5$$

Numerically verification with code using the expression $y \sim x1$, gives a linear regression coefficient of 1.510 (with a confidence region which includes the true value 1.5). The other expressions will not work.

c) Since there is a backdoor path between $x2$ and y, the expression needs to include the variables from a valid adjustment set. The expression $y \sim x1 + x2$ does this since $x1$ is $x2$'s only parent.

This can also be verified similiarly to how it was done in b). Now

$$y = (c41 + c43 \cdot c31)x1 + (c42 + c43 \cdot c32)x2 + bias + ny + c43 \cdot n3.$$

This means that the causal effect from $x2$ to y is $c42 + c43 \cdot c32 = 1.1$. This is also what the linear regression finds if the expression $y \sim x1 + x2$ is used. The other expressions will not work.

6 Estimation theory

The inequality comes from the CRLB Theorem (the bias free version)

$$\mathrm{Cov}(\hat{\theta}_N) \geq [\mathcal{I}(\theta)]^{-1}, \tag{1}$$

where $\mathcal{I}(\theta)$ is the Fisher information matrix and

$$\mathrm{Cov}(\hat{\theta}_N) = E\left[\left(\hat{\theta}_N - E\left(\hat{\theta}_N\right)\right)^2\right].$$

Since $\hat{\theta}_N$ is an unbiased estimate

$$\mathrm{Cov}(\hat{\theta}_N) = E\left[\left(\hat{\theta}_N - \theta\right)^2\right],$$

which is the left hand side of the inequality that we should show.

The Fisher information from independent samples is additive, meaning that

$$\mathcal{I}(\theta) = \sum_{n=1}^{N} \mathcal{I}_n(\theta). \tag{2}$$

The samples y_n are independent since e_n is white and since the term $\exp(\theta t_n)$ is non-random since t_n is measured. We have that

$$\mathcal{I}_n(\theta) = -E\left[\frac{d^2 l_n(y_n;\theta)}{d\theta^2}\right], \tag{3}$$

where $l_n(y_n;\theta)$ is the log-likelihood function

$$l_n(y_n;\theta) = log\left(p(y_n;\theta)\right),$$

where $p(y_n;\theta)$ is probability density function of y_n. Here y_n consists of a deterministic part, $\exp(\theta t_n)$, and one stochastic part, e_n. Since $e_n \in N(0,1)$ $y_n \in N(\exp(\theta t_n), 1)$, which means that

$$\begin{aligned}
&p(y_n;\theta) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{1}{2}(y_n - e^{\theta t_n})^2\right) \\
&\implies l_n(y_n;\theta) = -\log\sqrt{2\pi} - \frac{1}{2}(y_n - e^{\theta t_n})^2 \\
&\implies \frac{dl_n(y_n;\theta)}{d\theta} = -\left(y_n - exp(\theta t_n)\right)\left(-exp(\theta t_n)t_n\right) = exp(\theta t_n)t_n y_n - t_n exp(2\theta t_n) \\
&\implies \frac{d^2 l(y_n;\theta)}{d\theta^2} = t_n^2 y_n exp(\theta t_n) - 2t_n^2 exp(2\theta t_n)
\end{aligned}$$

Recalling equation (3) gives

$$\begin{aligned}
\mathcal{I}_n(\theta) &= -E\left[t_n^2 y_n exp(\theta t_n) - 2t_n^2 exp(2\theta t_n)\right] = -t_n^2 E[y_n] exp(\theta t_n) + 2t_n^2 exp(2\theta t_n) = \\
&= -t_n^2 exp(\theta t_n) exp(\theta t_n) + 2t_n^2 exp(2\theta t_n) = t_n^2 exp(2\theta t_n).
\end{aligned}$$

and then using (2) gives

$$\mathcal{I}(\theta) = \sum_{n=1}^{N} t_n^2 exp(2\theta t_n).$$

The CRLB theorem (1) therefore gives the wanted result

$$E(\hat{\theta}_N - \theta)^2 = \mathrm{Cov}(\hat{\theta}_N) \geq [\mathcal{I}(\theta)]^{-1} = \left(\sum_{n=1}^{N} t_n^2 \exp\left(2t_n\theta\right) \right)^{-1} \qquad \square$$