# Solutions to Assignment

Rongfei Jin

April 24, 2025

## Question 1

We use the PACF to determine the order of the AR model. The PACF is defined as the correlation between $Y_t$ and $Y_{t-k}$ after removing the effect of all the intermediate variables $Y_{t-1}, Y_{t-2}, \ldots, Y_{t-(k-1)}$.

Consider the $AR(p+1)$ model:

$$Y_t = a_0 + \sum_{k=1}^{p+1} a_k Y_{t-k} + \epsilon_t, \quad \epsilon_t$$

Since the true model is $AR(p)$, we have $a_{p+1} = 0$ and the PACF should be zero for all lags greater than $p$.

Now to determine the PACF, we can use the Yule-Walker equations, which relate the autocorrelations of the time series to the coefficients of the AR model and we can compute the significant levels to check if PACF is significantly different from zero.

See q1q2.py for the implementation.

# Question 2

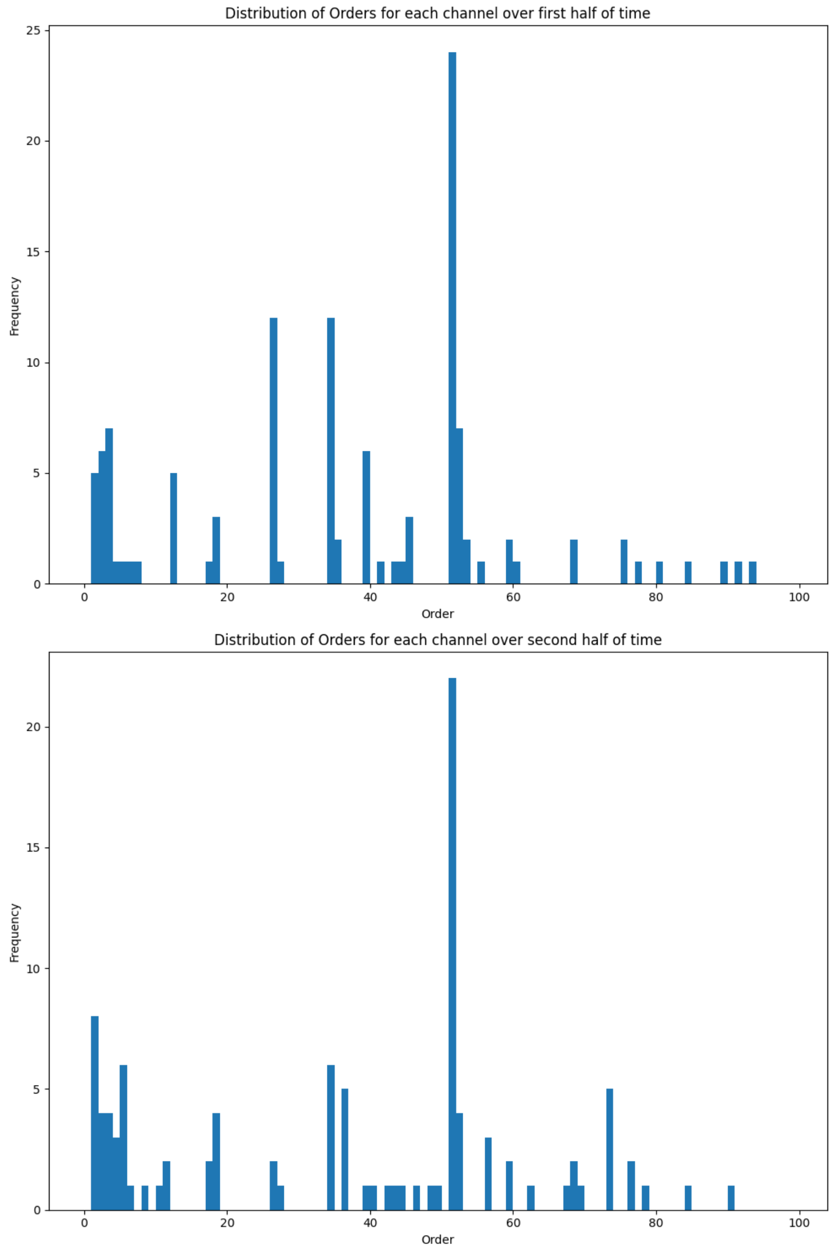


Figure 1: Distribution of Orders for each channel over 0-500ms, and $t_{onset}$ to $t_{onset}$+500ms

We plot 2 histograms, the first one is the distribution of orders for each channel over 0-500ms, and the second one is the distribution of orders for each channel over $t_{onset}$ to $t_{onset}$+500ms. The overall distribution of orders is similar, we have channels with estimated order around 50 but the second histogram has more channels with estimated order of 2. This suggests that the epileptic activity changes the brain signal and the effect is not immediate.

# Question 3

We first consdier the Autocovariance function

$$\gamma(h) = \mathrm{Cov}(y_t, y_{t-h}) = \mathrm{E}[y_t y_{t-h}] - \mathrm{E}[y_t]\,\mathrm{E}[y_{t-h}]$$

and the correlation function

$$\mathrm{Corr}(y_t, y_{t-h}) = \frac{\mathrm{Cov}(y_t, y_{t-h})}{\sqrt{\mathrm{Var}(y_t)\,\mathrm{Var}(y_{t-h})}}$$

The ACF is defined as the correlation between $y_t$ and $y_{t-h}$ after removing the effect of all the intermediate variables $y_{t-1}, y_{t-2}, \ldots, y_{t-(h-1)}$. We express the ACF as

$$\rho(h) = \frac{\gamma(h)}{\gamma(0)}$$

The PACF is defined as the correlation between $y_t$ and $y_{t-h}$ after removing the effect of all the intermediate variables $y_{t-1}, y_{t-2}, \ldots, y_{t-(h-1)}$. We express the PACF for a staionary time series as

$$\phi_{11} = \gamma(1)$$

and

$$\phi_{hh} = \mathrm{Corr}(y_{t+h} - \hat{y}_{t+h}, y_{t-h} - \hat{y}_{t-h})$$

where $\hat{y}_{t+h}$ and $\hat{y}_{t-h}$ are the predicted values of $y_{t+h}$ and $y_{t-h}$ using the regression model.

$$\hat{y}_{t+h} = \sum_{i=1}^{h-1} \beta_i y_{t+h-i}$$

$$\hat{y}_t = \sum_{i=1}^{h-1} \beta_i y_{t+i}$$

And the we can use the ACF and PACF to determine the order of the AR, MA, or ARMA model.

| Function | AR(p) | MA(q) | ARMA(p,q) |
|---|---|---|---|
| ACF | Tails off | Cuts off after lag q | Tails off |
| PACF | Cuts off after lag p | Tails off | Tails off |

# Question 4

We first derive the ACF for MA(1) model.

$$\begin{aligned}\gamma(1) &= \mathrm{Cov}(y_t, y_{t-1}) = \mathrm{Cov}(\epsilon_t - \theta_1\epsilon_{t-1}, \epsilon_{t-1} - \theta_1\epsilon_{t-2}) \\ &= 0 + 0 + \theta_1\sigma^2 + 0 = \theta_1\sigma^2 \quad \epsilon_k \text{ are independent}\end{aligned}$$

$$\gamma(0) = \mathrm{Var}(y_t) = \mathrm{Var}(\epsilon_t - \theta_1\epsilon_{t-1}) = \sigma^2 + \theta_1^2\sigma^2 = (1+\theta_1^2)\sigma^2$$

$$\rho(1) = \frac{\gamma(1)}{\gamma(0)} = \frac{\theta_1\sigma^2}{(1+\theta_1^2)\sigma^2} = \frac{\theta_1}{1+\theta_1^2}$$

We estimate the ACF using the sample ACF, and we solve the equation for $\hat{\theta}_1$

$$\hat{\theta}_1 = \frac{1 \pm \sqrt{1 - 4\hat{\rho}(1)^2}}{2\hat{\rho}(1)}$$

We then choose the solution $|\hat{\theta}_1| < 1$ for MA(1) model to be invertible, which is important for the equivalent AR model to be causal for further prediction.

See q4q5.py for the implementation.

# Question 5

The simulated data is actually MA(5) model, so both (i) and (ii) models are unable to capture the MA(5) model and produce a incorrect estimate of the parameters. By plotting the ACF and PACF, we can see that the ACF tails off and the PACF cuts off after lag 5, which is the order of the MA(5) model so with the proper order, we can estimate the parameters correctly.

See q4q5.py for the implementation.

# Question 6

We use the library pmdarima to fit the best ARMA models for each channel. We then plot the distribution of the orders for each channel over 0-500ms, and $t_{onset}$ to $t_{onset}$+500ms.

The two distributions exhibit a similar pattern, but the ictal period data seems to be better modeled with smaller MA order

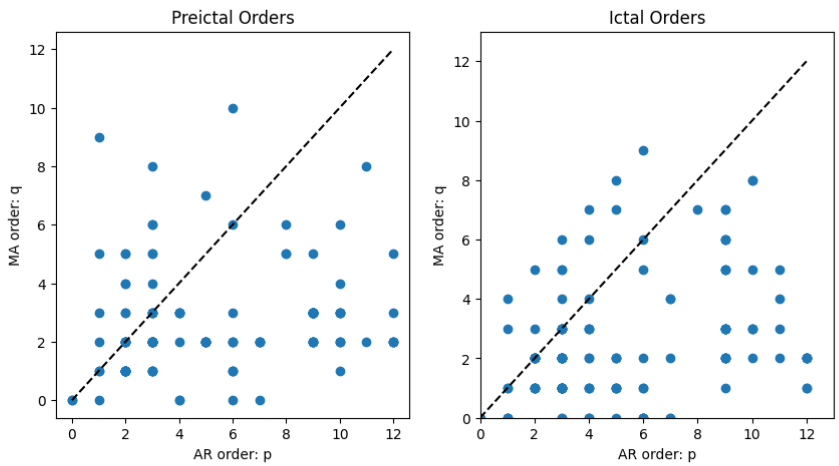


Figure 2: Distribution of Orders for each channel over 0-500ms, and $t_{onset}$ to $t_{onset}$+500ms

# Question 7

Note that $x_k, \mathbf{x_{-k}} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Theta}^{-1})$, we can partition the mean and covariance matrix as

$$\boldsymbol{\mu} = \begin{pmatrix} \mu_k \\ \boldsymbol{\mu}_{-k} \end{pmatrix}$$

$$\boldsymbol{\Theta} = \begin{pmatrix} \boldsymbol{\Theta}_{kk} & \boldsymbol{\Theta}_{k,-k} \\ \boldsymbol{\Theta}_{-k,k} & \boldsymbol{\Theta}_{-k,-k} \end{pmatrix}$$

Notice that the kernel of the joint distribution of $x_k$ and $\mathbf{x_{-k}}$ is

$$\exp\left(-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^T\Theta(\mathbf{x}-\boldsymbol{\mu})\right)$$

We can expand the quadratic form as

$$(\mathbf{x}-\boldsymbol{\mu})^T\Theta(\mathbf{x}-\boldsymbol{\mu}) = \Theta_{kk}(x_k-\mu_k)^2 + 2\Theta_{k,-k}(x_k-\mu_k)(\mathbf{x_{-k}}-\boldsymbol{\mu}_{\mathbf{-k}}) + (\mathbf{x_{-k}}-\boldsymbol{\mu}_{\mathbf{-k}})^T\Theta_{-k,-k}(\mathbf{x_{-k}}-\boldsymbol{\mu}_{\mathbf{-k}})$$

Now the conditional expectation of $x_k$ given $\mathbf{x_{-k}}$ is related to the $x_k$ terms in the kernel of the joint distribution of $x_k$ and $\mathbf{x_{-k}}$. We can rewrite the terms in the kernel as

$$-\frac{1}{2}\left((x_k-\mu_k)^2\Theta_{kk}+2(x_k-\mu_k)(\mathbf{x_{-k}}-\boldsymbol{\mu}_{\mathbf{-k}})\Theta_{k,-k}\right)$$

We can complete the square as

$$-\frac{1}{2}\Theta_{kk}\left((x_k-\mu_k+\frac{\Theta_{k,-k}}{\Theta_{kk}}(\mathbf{x_{-k}}-\boldsymbol{\mu}_{\mathbf{-k}}))^2\right)$$

Based on the kernel, we can derive the conditional mean as

$$\mathrm{E}[x_k|\mathbf{x_{-k}}]=\mu_k+\frac{\Theta_{k,-k}}{\Theta_{kk}}(\mathbf{x_{-k}}-\boldsymbol{\mu}_{\mathbf{-k}})$$

# Question 8

To estimate the $\boldsymbol{\Theta}$, first notice that the previous expectation shows a linear relationship between $x_k$ and $\mathbf{x_{-k}}$. We can rewrite the expectation as

$$\hat{x}_k=\mu_k+\boldsymbol{\beta}^T(\mathbf{x_{-k}}-\boldsymbol{\mu}_{\mathbf{-k}})+\epsilon_k$$

where $\boldsymbol{\beta}=\frac{\Theta_{k,-k}}{\Theta_{kk}}$ and $\epsilon_k\sim\mathcal{N}(0,\frac{1}{\Theta_{kk}})$.

Now we can use the lasso regression to estimate the $\boldsymbol{\Theta}$ based on the linear relationship.

We let $\hat{\mu}_k=x'_k$ and $\hat{\boldsymbol{\mu}}_{-k}=\mathbf{x}'_{-k}$ where $x'_k$ and $\mathbf{x}'_{-k}$ are the sample data, we estimate the $\frac{1}{\Theta_{kk}}$ as the inverse of the variance of the residuals.

$\Theta_{kk}=\frac{\beta}{var(\epsilon_k)}$

See q8.py for the implementation.

# Question 9

We first write the pdf of multivariate normal distribution as

$$f(\mathbf{x})=(2\pi)^{-\frac{n}{2}}\det(\boldsymbol{\Sigma})^{-1/2}\exp\left(-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^T\boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})\right)$$

The log likelihood function is

$$\log L(\boldsymbol{\mu},\boldsymbol{\Sigma})=-\frac{n}{2}\log(2\pi)-\frac{1}{2}\log(\det(\boldsymbol{\Sigma}))-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^T\boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})$$

By using the determiant property, we can rewrite the log likelihood function as

$$\log L(\boldsymbol{\mu},\boldsymbol{\Sigma})=-\frac{n}{2}\log(2\pi)+\frac{1}{2}\log(\det(\boldsymbol{\Theta}|))-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^T\boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})$$

Notice that the last term is a scalar, so we can use associative properties of trace to rewrite the log likelihood function as

$$\log L(\boldsymbol{\mu},\boldsymbol{\Theta})=-\frac{n}{2}\log(2\pi)+\frac{1}{2}\log(\det(\boldsymbol{\Theta}))-\frac{1}{2}\operatorname{tr}(\mathbf{S}\boldsymbol{\Theta})$$

We now omit the constant term since we only consider the maximization with respect to $\boldsymbol{\Theta}$.

$$\log\det(\boldsymbol{\Theta})-\mathbf{S}\boldsymbol{\Theta}$$

Finally we can add the penalty term to the log likelihood function to get the penalized log likelihood function.

$$\log\det(\boldsymbol{\Theta})-\mathbf{S}\boldsymbol{\Theta}+\rho\|\boldsymbol{\Theta}\|_1$$

# Question 10

The difference between the two methods is the first method is not based on the maximum likelihood estimator of the precision matrix, while the second method is and uses coordinate descent to estimate the precision matrix.

One advantage of the first method is that can be easily run in parallel, while the second method needs to compute the whole matrix. One limitation of the first method is that it may not produce a positive definite matrix. So a good senario for the first method is when the sample size is large and the sample precision matrix is close to the true precision matrix.

See q10.py for the implementation.

# Question 11

We use the same architecture of RNN to predict the epileptic activity for preictal and ictal data and compare the performance of the two models in terms of MSE. The preictal model has a lower MSE than the ictal model. The fact that the same architecture of RNN is not able to model the preictal and ictal data with the same error after cross validation, suggests that the preictal and ictal data have different characteristics. Intuitively, the models with the same complexity should be able to model the preictal and ictal data with the same error. The difference in performance may be due to the difference in the distribution of the data in different channels.

See q11.py for the implementation.

# Question 12

See q12.py for the implementation.

# Question 13

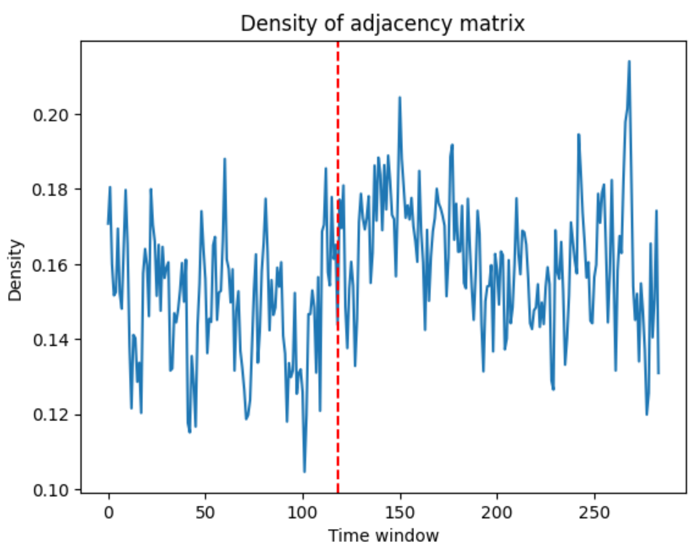


Figure 3: Distribution of density

The plots suggests a potential trend in the data. Here we use the chow test to test the null hypothesis that the two regression model on data before and after are the same. The test gives a p-value of 3e-6, which is very small, so we reject the null hypothesis and conclude that there is a trend in the data. Although chow test is not the best test for this purpose, it is still a good test to detect the structural change of the data.

See q12q13.py for the implementation.

# Question 14

Here we use the RNN again but this time we try to classify time windows into preictal and ictal states. RNN takes a sequence of data as input and output a probability distribution of the class of the input. The difference here, compared to previous RNN is that we use sigmoid function to output the probability of the class. We then compute the ROC AUC and f1 score to evaluate the performance of the model.

The distibution of the density is shown in the following figure. Some channels achieves a high f1 score and ROC AUC, but some channels achieve a low f1 score and ROC AUC. The difference in performance can be due to the difference in the distribution of the data in different channels. What's more, the performance of the model may be directly related to the spatial characteristics of the electrodes. Electrode with higher scores may be closer to the seizure onset or can be more likely to be affected by the seizure. An area of interest analysis may be helpful to understand the spatial characteristics of the electrodes.

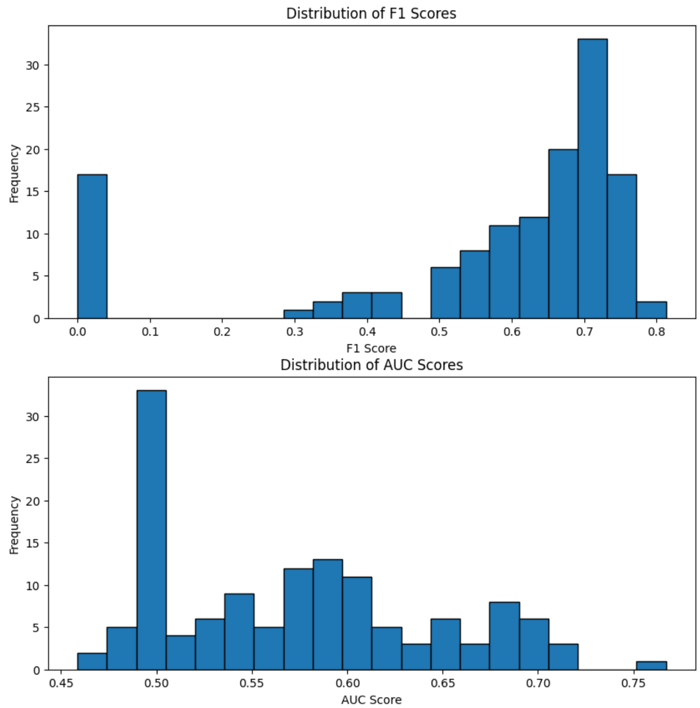


Figure 4: Distribution of density

See q14.py for the implementation.