Speech Recognition using ARMA Model & Levenberg-Marduard Algorithm

Reza Jafari[1] and Amir H. Jafari[2]

[1] Virginia Tech University, Falls church VA 22043, USA,
rjafari@vt.edu
[2] George Washington University , Washington, DC 20052, USA,
ajafari@gwu.edu

Abstract. Autoregressive Moving Average (ARMA) is a simple linear model with memory that can be used for speech recognition problems. This is why, this paper utilized the derivation of ARMA model for the speech recognition. The flexibility of ARMA model helps in derivation of an accurate model that recognizes the pronunciation of letter **B**. The Generalized Partial Autocorrelation (GPAC) analysis has been used for the preliminary identification and the Maximum Likelihood Estimator (Levenberg-Marduardt) is used for the parameter estimations. Several models have been developed to recognize the letter B that are pronounced by a lady 30 times. The simplest model has been chosen at the end. The accuracy of the final model has been checked using χ^2 test.

Keywords: speech recognition, nonlinear optimization, residual analysis

1 Introduction

Modeling is a key step in analyzing every science problems. Having an accurate model of a phenomena will help us to better understand the phenomena and helps us to make a prediction on unseen observations. In statistical modeling, model is developed using the observed data. The developed model then needs to be verified to make sure it is not over fitted and also generalized ([11], [9]). Speech recognition is a challenging modeling problem because it is a time series data. In the time series data set, sequences needs to be fed to the model for system identification. Autocorrelation function and generalized autocorrelation functions are usually used for time-series data set ([3], [7]). The general method for identifying time-series data is to perform preliminary order determination. Box-Jenkins [1] introduced the Autocorrelation and partial Autocorrelation (ACF/PAC) approach as the most effective way for estimating the order of the model in the time domain. The order determination can also be perform using the frequency approach and the power spectrum analysis as discussed in [1].

Time series data can also be modeled using nonlinear dynamic network i.e. recurrent neural network as discussed in [5]. Nonlinear dynamic neural networks are powerful but they suffer from stability issue. Due to the inherent feedback connection and nonlinear activation functions the stability analysis of these network is challenging. In [13], [12] global stability of the nonlinear dynamic networks are discussed using the method of dissipativity domain approach. The Autoregressive Moving average (ARMA) is flexible linear model that can be used for ordered data similar to speech recognition. In this research, the ARMA model is used for speech recognition.

The ARMA model is a conventional method in classification problems. The modern method is neural network and deep learning. The famous complex neural network model for classification is Long-Short-Term-Memory (LSTM). The advantage of ARMA model compared to the complicated neural network model is the simplicity which makes these kind of models less prone to over fitting.

The general overview of this paper is given in section 2. In section 3 a brief history on the data used in this project is provided. Section 4 gives the description of the type of model which is used and other types of models that are to be considered. Section 5 is the core part of this report. In this section we explained all the steps in finding the final model. In this section two different models are developed. Section 6, use the simplest model for this speech recognition problem. A brief summary and conclusion is given in section 7.

2 PROJECT OVERVIEW

The overall view of the project depicted in Figure(1). The flowchart shows all the procedure to obtain the final model for the given data. At the beginning we start off with the initial guess by looking at the GPAC table of the original data. Then we implement Leunbrg-Marrqurrdt Algorithm to estimate the parameters. If the residual is white then we need to check whether a_{na} and b_{nb} are close to zero or not (lie between ± 2 standard deviation). If they are close to zero then we will ignore them (path M). If No then we need to check for zero-pole cancelation. If there is no zero-pole cancelation then we are done. If there is zero-pole cancelation then we will decrease the order by the number of common zeros and common poles (path N). If the diagnostic test (Chi-Square test) fails then we need to adjust the order. This job usually accomplish by increasing the model order by the pattern observed from the GPAC table of the residual. We continue this procedure until reaching to the best possible model. There are several factors which we may need to consider during this procedure. Our objective here is to make the residual as white as possible. The test for measuring the whiteness of residual is Chi-Square test. Meanwhile we should always consider the sum square error, the variance of residual and the number of parameters. The smaller the variance of the residual the better but we don't forget that by increasing the number of parameters in the model the variance always goes down. Also having a model with too many parameters is not preferable. As we will mention in section 5 and 6 our decision for choosing the final model is not based on only one factor. There are several factors combine together to choose the best model.

3 DATA DESCRIPTION

The data which we used for this projet is a speech signal generated by a lady working at American University of Sharjah. She was asked to pronounce the letter ***B*** for 30

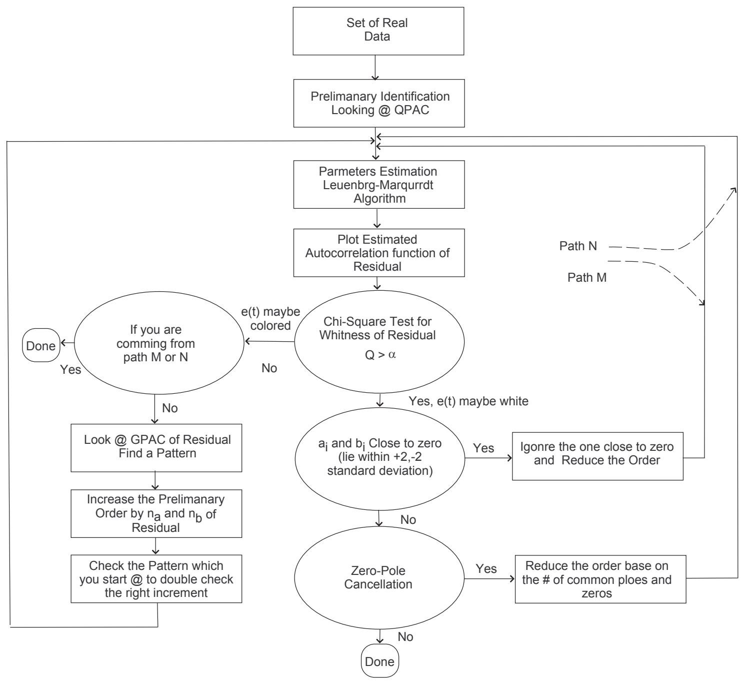

Fig. 1: Project Framework

times. Figure (2) shows the sampled data. Since the project was defined to find a simplest linear model for this set of data so the part of the data which was approximately stationary were chosen. Figure (4) shows the auto correlation function of selected part of the data. In order to verify whether the selected part of the data is possible to be fit into a linear model, we did two tests. First the Auto correlation function should decay and second if you look at the plot of the data in normal view and the upside down view it should not look very different. Although the selected data passed these two tests but as we shall see later we could not find a linear model which could fully represent this part of data. The reason of this issue will be reviewed in details in the summary and conclusion section.

4 LINEAR MODELS

There are several linear models which can be used for modeling.(i.e ARMA, ARX, ARMAX,...). The strategy on choosing the model depends on the application which we are not going to discuss in the report (refer to [7]). We will compare two different models (ARMA and ARMAX) against each other and give the related equations.

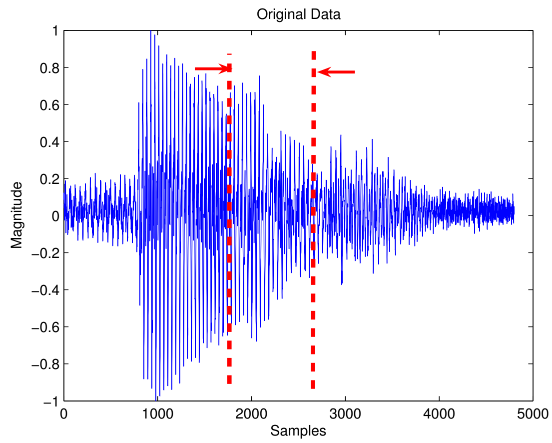

Fig. 2: Original data

Auto-Regressive Moving-Average (ARMA) process can be represented as

$$\begin{aligned} y(t) + a_1 y(t-1) + ... + a_{n_a} y(t-n_a) = e(t) + ... \\ b_1 e(t-1) + ... b_{n_b} e(t-n_b) \end{aligned} \tag{1}$$

where $e(t)$ is white noise. The other representation of this process can be written as

$$y(t) = R(q)e(t)$$

where $R(q) = \frac{B(q)}{A(q)}$ and

$$\begin{aligned} B(q) = 1 + b_1 q^{-1} + ... + b_{n_b} q^{-n_b} \\ A(q) = 1 + a_1 q^{-1} + ... + a_{n_a} q^{-n_a} \end{aligned}$$

If $n_b = 0$, we have autoregressive(AR) model

$$y(t) + a_1 y(t-1) + ... + a_{na} y(t-n_a) = e(t) \tag{2}$$

And if $n_a = 0$, we have moving average (MA) model:

$$y(t) = e(y) + b_1 e(t-1) + ... b_{n_b} e(t-n_b) \tag{3}$$

ARMAX is another linear model which is more general than ARMA model. The ARMAX model can be represented as

$$\begin{aligned} y(t) + a_1 y(t-1) + ... + a_{n_a} y(t-n_a) = u(t) + ... \\ b_1 u(t-1) + ... b_{n_b} u(t-n_b) + ... \\ e(t) + c_1 e(t-1) + ... c_{n_c} e(t-n_c) \end{aligned} \tag{4}$$

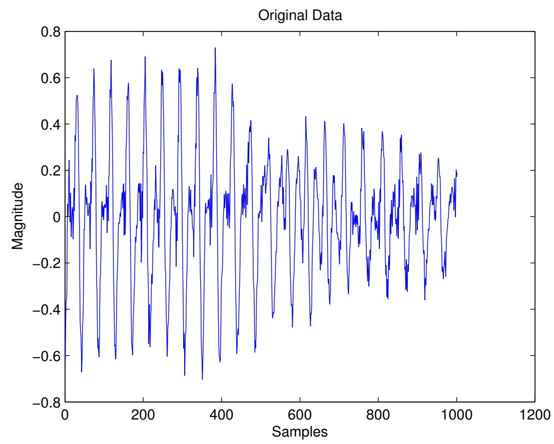

Fig. 3: Selected part of the Data

where $u(t)$ is exogenous input and $e(t)$ is white noise. If we compare this model against ARMA model we can see that ARMAX is more general in the sense that we are considering an exogenous input $u(t)$. For the purpose of this course we used the ARMA model to fit the given data.

5 MODEL DEVELOPMENT

In this section we will investigate all the procedure to obtain the simplest possible model. This section of the report can be considered as the core part of this project.

5.1 PRELIMINARY IDENTIFICATION

As we mentioned earlier the first step in finding the model of a given data is a initial guess on the order of the model. This job can be done by looking at the GPAC of original data. The GPAC of original data is given in Table (1). From the table at $k = 2$ and $j = 3$ we can see a pattern. Based on this pattern we may consider an initial model to be an ARMA(2,3). The next step is to estimate the parameters.

5.2 PARAMETERS ESTIMATION AND DIAGNOSTIC TEST

In this section we will estimate the parameters for the initial guess which we made in the previous section. The program for this estimation is based on *Leunbrg-Marrqurrdt Algorithm* which is based on Maximum Likelihood Estimation (MLE). There are three input arguments for this program (na,nb and $y(t)$). The program defined a performance index $J(k)$ and compute the gradient of residual and update the parameters in order to minimize the sum square error. There is a subroutine in the code which evaluate the

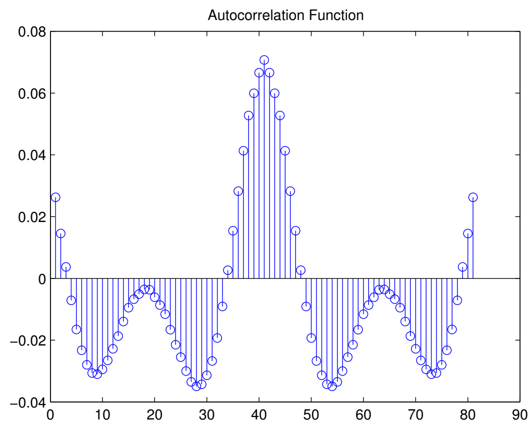

Fig. 4: Initial Auto Correlation Function

Table 1: GPAC of Original Data

	k = 1	k = 2	k = 3	k = 4	k = 5	k = 6	k = 7	k = 8	k = 9	k = 10
j =0	0.9415	-0.3449	-0.0246	-0.6423	-0.0183	-0.2578	0.2377	-0.0518	0.0955	0.1492
j =1	0.8999	-0.4041	8.9768	-0.6419	9.0286	-0.2736	0.1848	0.3857	0.1756	0.1758
j =2	0.8800	-3.8320	-1.5967	-0.5609	-0.5875	-0.6461	0.3120	-0.3253	0.4995	0.1599
j =3	0.7840	-0.7215	-0.0109	-0.3501	0.0695	-0.2598	0.7758	0.5577	0.3074	0.6502
j =4	0.6835	-0.7312	22.9645	-0.3509	-1.3933	-0.1356	0.4981	-0.6030	-0.1462	0.2719
j =5	0.5464	-1.0177	-0.5437	-0.1386	-0.6747	-2.5815	0.4394	-0.8544	-1.8009	0.3563
j =6	0.1729	-0.9033	-0.2780	1.4622	-0.4011	-0.5105	-0.1528	-0.4683	0.1118	-0.8331
j =7	-3.3990	-0.9093	-2.5992	0.5814	-1.1557	-0.4143	1.9563	-0.4847	-3.5943	-0.8738
j =8	2.1281	-0.9736	-0.9011	-2.5310	-0.7637	-1.4436	0.5391	0.0259	0.3126	-0.3772
j =9	1.3842	-0.8342	0.8853	-1.0754	0.8592	-0.7349	0.5893	-6.8364	0.3465	-0.0185

whiteness of residual by checking the Chi-Square Test.

Remark: The first coefficient in AR and MA part of ARMA model Eq.(1) is always 1, so the program ignored these coefficients and estimate the other parameters. Figure(5) shows the performance index ($J(k)$) and the Autocorrelation function of the residual for the case $na = 2$ and $nb = 3$.

5.3 CHI-SQUARE TEST FOR WHITENESS

Let $\hat{R}_e(\tau)$ and $r_e(\tau)$ be a estimated and normalized autocorrelation function of residual so

$$r_e(\tau) = \frac{\hat{R}_e(\tau)}{\hat{R}_e(0)} \tag{5}$$

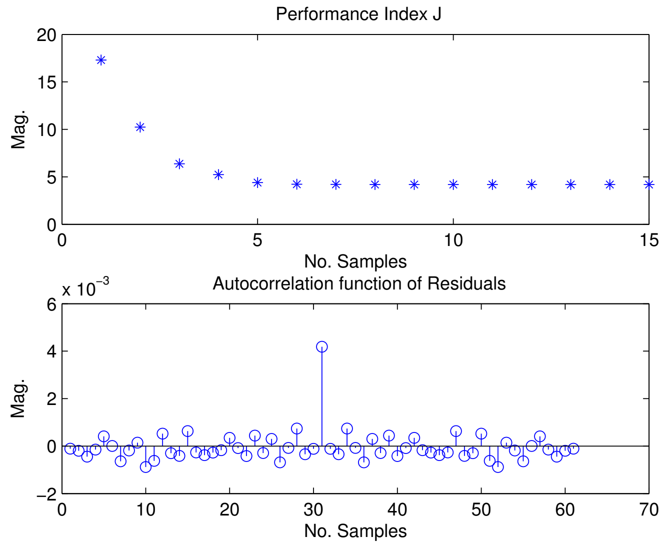

Fig. 5: Performance Index and A.C.F of Residual[$n_a = 2, n_b = 3$]

and Q be defined as

$$Q = N \sum_{\tau=1}^{k} r_e^2(\tau) \tag{6}$$

where N is the number of data points and k is the number of lags to be considered (usually $20 \leq k \leq 40$). If $e(t)$ is white noise sequence then Q will be distributed as $\mathcal{X}^2(DOF)$ where

$$DOF = k - n_a - n_b$$

and $\mathcal{X}$ is a Chi-Square variable. If $e(t)$ is not white then $Q > \alpha$ and it is called Q is *inflated*. Given the desired degree of freedom and the confidence rate (ϵ) we can read the α value from the Chi-Square table.

There are two types of error associated with this test:

- ***Error Type I***: This is an error when $e(t)$ is not white but $Q < \alpha$.
- ***Error Type II***: This is an error when $e(t)$ is in fact white but $Q > \alpha$.

There is no way that we can overcome these types of errors, because Chi-Square test is an *statistical test*. By changing the value of ϵ we can increase or decrease the chance of one type of error, but we should always keep in mind that there is a trade-off between these two types of errors. In this project we considered $\epsilon = 0.05$ and $k = 30$.

Figure(6) shows the density function of Q and the region where $e(t)$ is white. The shaded region ($Pr(Q > \alpha)$) is the place where $e(t)$ is probably *colored* and the region where $Pr(Q < \alpha)$ is the place where $e(t)$ is probably *white*.

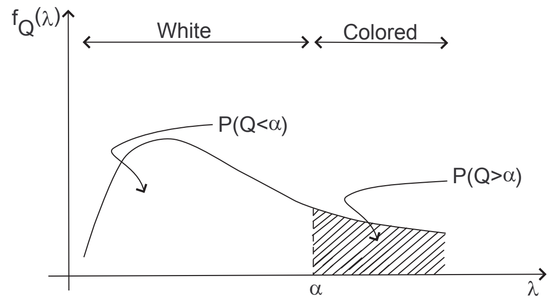

Fig. 6: Density function of Q

Considering above background on Chi-Square test we can proceed on the diagnostic section of the program. Table(2) shows the results of Chi-Square test for the case when $na = 2$ and $nb = 3$. It is clear that the test failed because $Q \gg \alpha$ so $e(t)$ is not white and we need to adjust the order.

Table 2: Estimation Results

$na = 2$, $nb = 3$	
Sum Square Error	4.1868
σ_e^2	0.0042
α	37.6525
Q	294.0110

For the proper adjustment of the order we need to look at the GPAC of residual and try to find an order. The GPAC of residual for $na = 2$ and $nb = 3$ is given in the Table(3). From this table we could see several patterns, such as $\{n_{ae} = 6,\ n_{be} = 1\}$, $\{n_{ae} = 5,\ n_{be} = 5\}$, $\{n_{ae} = 9,\ n_{be} = 0\}$ or $\{n_{ae} = 3, n_{be} = 0\}$. Among all the observed patterns the pattern $k = 5$ and $j = 5$ has been chosen because it gave us less square error. Then we add the initial order with the residual order and run the program for another estimation.

The results of the new estimation and the plot of $\hat{R}(\tau)$ and $J(k)$ is given in the Table (4) and Figure(7). From Table(5) it is clear that we are moving in a right direction because the first row of GPAC table became smaller.

From Table (4) is clear that Q is inflated and the Chi-Square test failed so we need to re-adjust the order by looking at the GPAC of residual.

By looking at the Table(5) we could see the following patterns:$\{n_{ae} = 2,\ n_{be} = 0\}$,$\{n_{ae} = 3,\ n_{be} = 0\}$,$\{n_{ae} = 4,\ n_{be} = 1\}$,$\{n_{ae} = 5,\ n_{be} = 0\}$,... . Among all the observed patterns the pattern $k = 4$ and $j = 1$ has been chosen for the same reason mentioned in the previous batch. Following the same procedure we increase the

Table 3: GPAC of Residual [$n_a = 2$, $n_b = 3$]

	k = 1	k = 2	k = 3	k = 4	k = 5	k = 6	k = 7	k = 8	k = 9	k = 10
j=0	-0.0273	-0.0830	0.1727	-0.0175	-0.1424	0.0379	-0.0886	0.1728	-0.1472	0.0095
j=1	3.0156	-0.1393	0.1645	-1.4209	-0.1469	-0.2948	-0.0154	0.0995	-0.1363	0.5992
j=2	-2.1422	-2.5562	0.1076	-0.0576	-0.2383	-0.2276	-1.5950	0.0906	-0.1039	-0.0564
j=3	-0.1013	-0.9912	-0.3652	-0.4850	-0.2725	0.1534	-0.2651	-0.0732	-0.1156	0.0202
j=4	9.2182	-0.9684	5.4170	-2.0081	-0.2334	-0.6351	-0.3005	0.6834	-0.1141	8.0792
j=5	-0.4435	0.2208	-0.2661	-0.0312	0.5522	-0.0677	-0.1079	0.1649	0.4906	0.1549
j=6	-0.9653	-0.4356	-0.2932	-4.6704	0.5394	-0.8545	-0.1742	0.4085	0.2148	0.8989
j=7	-1.4962	1.4963	-1.0009	0.3679	0.3670	0.4238	1.0559	0.2733	-0.7467	0.4921
j=8	-0.9462	-2.9428	-2.3282	1.4450	0.2312	-0.9442	0.2816	1.2285	0.0597	0.1080
j=9	0.1972	-0.7221	0.0938	0.3542	2.4144	-0.7217	3.3026	1.2763	-2.3402	0.2294

Table 4: Estimation Results

$n_a = 7$, $n_b = 8$	
Sum Square Error	3.5941
σ_e^2	0.0036
α	24.9958
Q	120.2972

previous order by 4 and 1 so we will have an ARMA(11,9) model. Let run the program with the new order and re-estimate the parameters and see the results of diagnostic test.

The results of the new estimation ($na = 11$ and $nb = 9$) and the plot of $\hat{R}(\tau)$ and $J(k)$ is given in the Table (6) and Figure(8).

From Table (6) it is clear that Q is inflated and Chi-Square test did not pass. We continued this procedure and we observed that even at higher order the Chi-Square test did not pass. This gave us an indiction that the selected data can not be fit into a linear model. Since the data was real (obviously nonlinear) so it is not very surprising that we are unable to fit it to a linear model. But as we can see from Table(7) the first row is very small(≈ 0). This is an indication that the residual is close to white (see Figure(8)). Since the order increment does not helpful in our case so we stop increasing the order and in the next section we will go through path M and N of the flowchart (Figure 1) to find out the final model.

5.4 FINAL MODEL DERIVATION

Path M and N of the flowchart is a test for the possibility of simpler model. In these two paths we investigate the possibility of falling inside ± 2 standard deviation for the last coefficients and also check the possibility of zero-pole cancelation. If ± 2 standard deviation of parameters contain zero so that coefficient may be ignored because it has a tiny effect in the process. Also the order of the model can be reduced by the common number of zeros and poles. This is due to the fact the common zeros and poles in transfer function will cancel each others during the process so they do not have any effects in the model.

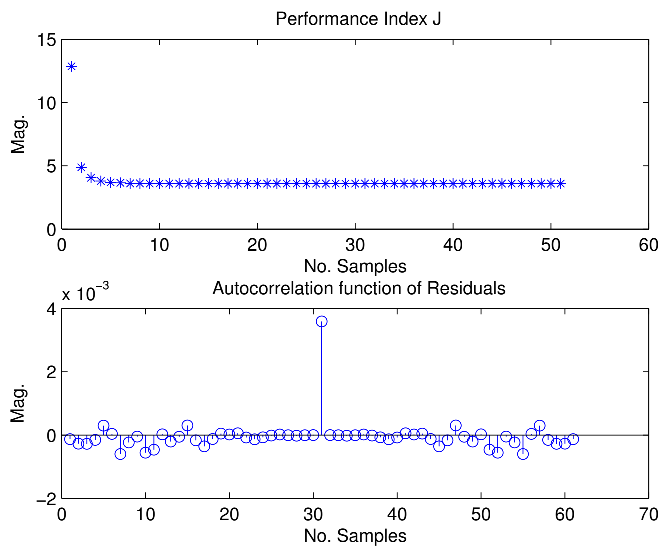

Fig. 7: Performance Index and A.C.F of Residual[$n_a = 7, n_b = 8$]

ARMA(11,9) was the order which we derived in the pervious section. Checking zeros and poles of the process, we observed the exitance of four possible common zeros and poles. The list of zeros and poles is as follow:

$$
\begin{aligned}
z_1 &= -0.7367 + 0.5340i & p_1 &= -0.6748 + 0.6300i \\
z_2 &= -0.7367 - 0.5340i & p_2 &= -0.6748 - 0.6300i \\
z_3 &= -0.6945 & p_3 &= -0.5566 \\
z_4 &= 0.3180 + 0.9268i & p_4 &= -0.0982 + 0.7556i \\
z_5 &= 0.3180 - 0.9268i & p_5 &= -0.0982 - 0.7556i \\
z_6 &= 0.8760 + 0.4479i & p_6 &= 0.3537 + 0.9153i \\
z_7 &= 0.8760 - 0.4479i & p_7 &= 0.3537 - 0.9153i \\
z_8 &= 0.3853 + 0.4582i & p_8 &= 0.8887 + 0.4278i \\
z_9 &= 0.3853 - 0.4582i & p_9 &= 0.8887 - 0.4278i \\
& & p_{10} &= 0.9262 + 0.2238i \\
& & p_{11} &= 0.9262 - 0.2238i
\end{aligned}
\tag{7}
$$

Table 5: GPAC of Residual [$n_a = 7$, $n_b = 8$]

	k = 1	k = 2	k = 3	k = 4	k = 5	k = 6	k = 7	k = 8	k = 9	k = 10
j=0	-0.0001	-0.0023	-0.0064	-0.0023	0.0051	-0.0050	-0.0203	-0.0373	-0.0225	0.0143
j=1	17.3401	-0.0020	-0.0056	-0.0167	0.0029	-0.0259	-0.0112	-0.0251	-0.0462	0.0210
j=2	2.7506	-7.8370	-0.0048	-0.0105	-0.0617	-0.0206	0.0149	-0.0184	-0.0342	0.0583
j=3	0.3534	-1.0607	2.2953	-0.0001	0.0168	-0.0297	-0.0445	-0.0258	-0.0423	-0.0430
j=4	-2.2625	-0.3994	2.2723	364.7433	0.0166	-0.0186	0.0243	-0.0551	-0.0486	0.2796
j=5	-0.9556	-8.4233	2.9141	5.1366	5.7426	-0.0105	-0.0409	-0.0308	-0.0016	0.0344
j=6	4.1359	-1.7822	-1.2792	1.9246	2.5066	-9.8114	-0.0198	-0.0298	-0.6252	0.0313
j=7	1.8400	-4.1228	-5.1377	2.9664	6.7516	-5.0781	7.6204	-0.0168	-0.0572	-0.0091
j=8	0.5962	-1.1115	2.0422	-3.3634	2.9732	-0.1903	3.2866	-11.2787	-0.0547	-0.1758
j=9	-0.6641	-0.3114	-0.5160	0.4887	2.8527	49.2425	2.9153	-0.6107	1.4903	0.1554

Table 6: Estimation Results

$n_a = 11$, $n_b = 9$	
Sum Square Error	3.4304
σ_e^2	0.0035
α	18.3070
Q	112.5791

Since

$$z_6 \approx p_8$$
$$z_7 \approx p_9$$
$$z_4 \approx p_6$$
$$z_5 \approx p_7$$

so they may be considered as potential common zeros and poles and they may cancel each other during the process. Considering above zero-pole cancelation the ARMA(11,9) model derived in previous section will reduce to ARMA(7,5).

Now we need to check whether the coefficients are close to zero (lie inside ± 2 standard deviation) or not. From (8) we can see that the coefficients are not small to be ignored.

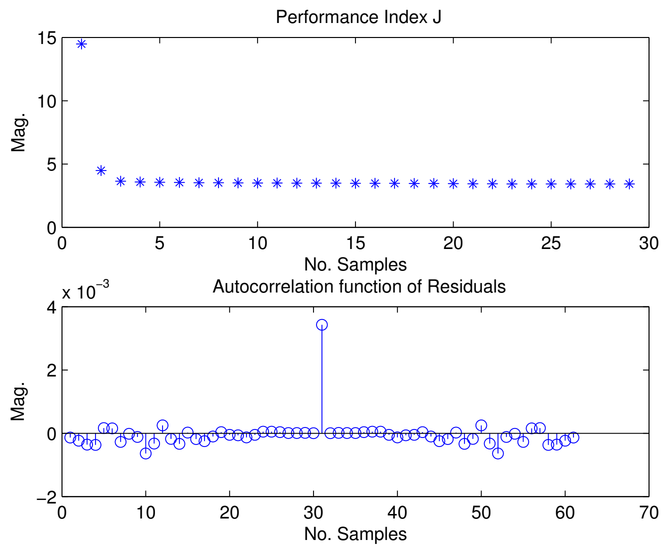

Fig. 8: Performance Index and A.C.F of Residual[$na = 11, nb = 9$]

$$
\begin{array}{llll}
\hat{a}_1 = -0.6689 & \sigma_{\hat{a}1} = 0.1848 & \hat{b}_1 = 0.6052 & \sigma_{\hat{b}1} = 0.1850 \\
\hat{a}_2 = -0.3786 & \sigma_{\hat{a}2} = 0.1632 & \hat{b}_2 = -0.3382 & \sigma_{\hat{b}2} = 0.1134 \\
\hat{a}_3 = -0.5773 & \sigma_{\hat{a}3} = 0.0769 & \hat{b}_3 = -0.3188 & \sigma_{\hat{b}3} = 0.0877 \\
\hat{a}_4 = 0.7865 & \sigma_{\hat{a}4} = 0.1105 & \hat{b}_4 = -0.4638 & \sigma_{\hat{b}4} = 0.0506 \\
\hat{a}_5 = -0.0629 & \sigma_{\hat{a}5} = 0.1059 & \hat{b}_5 = 0.2261 & \sigma_{\hat{b}5} = 0.0929 \\
\hat{a}_6 = 0.2878 & \sigma_{\hat{a}6} = 0.0780 & & \\
\hat{a}_7 = -0.2078 & \sigma_{\hat{a}7} = 0.0753 & &
\end{array}
\tag{8}
$$

Considering above arguments we end up with the best (simplest) model representing the given data. This model is ***ARMA(7,5)***.

Table(8) and (9) shows the results of simulation and the GPAC of residual for the final model. The autocorrelation function of residual and performance index $J(k)$ plotted in Figure(9)

5.5 ALTERNATIVE MODEL

In the section 5.1 we considered the initial model to be ARMA(2,3), but this is not the only possible choice. By looking at Table(1) we can see other patterns such as ARMA(2,0), ARMA(4,0), ARMA(4,3)...

Table 7: GPAC of Residual [$n_a = 11$, $n_b = 9$]

	k = 1	k = 2	k = 3	k = 4	k = 5	k = 6	k = 7	k = 8	k = 9	k = 10
j=0	0.0008	0.0032	0.0022	0.0015	0.01	0.0148	0.0148	-0.0144	-0.0387	0.0192
j=1	3.9015	0.0026	0.0000	-0.0128	0.0077	0.0047	0.0293	-0.0542	-0.0315	0.0108
j=2	0.6969	0.0014	7.4914	-0.01287	0.0095	-0.0272	0.0207	-0.0074	-0.035	-0.1174
j=3	0.6986	-1566.3	3.3392	-0.0019	0.0183	0.0214	0.0178	-0.0916	-0.0377	-0.0566
j=4	6.3322	3.8117	2.8362	26.953	0.0163	-0.0058	0.0331	-0.0083	-0.0484	0.1009
j=5	1.4785	-0.9114	-2.6209	3.0566	1.1161	0.11109	0.0343	-0.1828	-0.0435	-0.0284
j=6	1.0059	-6.1794	-4.6662	3.9051	21.642	-6.761	0.0117	-0.012	-0.0313	0.0763
j=7	-0.957	-1.7892	-0.6305	-3.2732	-0.79457	-4.4279	-4.526	-0.0406	-0.0395	-0.0475
j=8	2.6897	-1.5495	7.4022	-2.9918	18.115	-4.1476	10.799	-10.514	-0.052	0.05919
j=9	0.4954	0.17224	-0.5759	0.86101	1.8135	-1.1617	-2.8737	3.3449	3.8087	-0.1348

Table 8: Estimation Results

$na = 7$, $nb = 5$	
Sum Square Error	3.6685
σ_e^2	0.0037
α	28.8693
Q	149.0728

For the alternative model we started with ARMA(4,0) (because it is the second simplest order) with sum square error equal to 4.3. Since this model did not pass the diagnostic test we looked at GPAC of residual and increased the order by $6, 6$. The sum square error for ARMA(10,6) was equal to 3.5. Again the diagnostic test failed and we increased the order by $2, 0$. The sum square error for ARMA(12,6) was 3.4. We stopped at this order because increasing the order gave us tiny difference in sum square error. In addition the first row in autocorrelation of residual was close to zero which was another indication to stop incrementing the model order. The diagnostic test did not pass in any order for the same reason which we mentioned in the previous section. Considering the standard deviation and zero-pole cancelation arguments the final reduced order for this case derived to be ***ARMA(11,5)***

Since the procedure in obtaining the final model order gave in details in previous section including all the Figures and Tables so for saving space we only provided the

Table 9: GPAC of Residual [$na = 7$, $nb = 5$]

	k = 1	k = 2	k = 3	k = 4	k = 5	k = 6	k = 7	k = 8	k = 9	k = 10
j=0	0.0021	0.0058	-0.0033	-0.0142	-0.0114	0.0042	0.0356	-0.0269	-0.0869	-0.0289
j=1	2.7465	0.0070	-0.0282	-0.0115	-0.0166	0.1012	0.0387	-0.1419	-0.0780	-0.2072
j=2	-0.5661	-2.2897	-0.0066	-0.0040	0.0156	-0.0463	0.0059	-0.0507	-0.0396	-0.0324
j=3	4.3124	-1.7526	1.0393	-0.0211	-0.0073	-0.0435	-0.3308	-0.0537	-0.0191	0.0145
j=4	0.8100	-1.1528	-4.5718	1.5261	0.1778	-0.0400	0.0706	-0.0605	-0.0403	0.0461
j=5	-0.3460	-2.2521	-6.6494	40.5909	8.9269	-0.0150	0.0289	-0.0545	0.0370	-0.0321
j=6	8.9433	-3.2263	-0.4017	3.1580	5.5646	10.7124	0.0122	-0.0526	-0.0538	-0.0395
j=7	-0.7417	-2.7447	-24.2339	3.8107	3.2748	6.1782	26.5979	-0.0530	0.0408	-0.2862
j=8	3.2657	-1.7670	1.3724	-0.4938	0.3163	0.2109	-0.1732	-0.2275	-0.6127	-0.3396
j=9	0.3422	-0.8636	0.7006	0.4993	0.6137	0.4618	-0.4519	-1.7624	1.4051	-0.2493

final order results. The estimated coefficients are as follow

$$
\begin{array}{llll}
\hat{a}_1 = -0.8136 & \sigma_{\hat{a}1} = 0.1760 & \hat{b}_1 = 0.4593 & \sigma_{\hat{b}1} = 0.1770 \\
\hat{a}_2 = 0.5823 & \sigma_{\hat{a}2} = 0.1288 & \hat{b}_2 = 0.4872 & \sigma_{\hat{b}2} = 0.1304 \\
\hat{a}_3 = -0.8690 & \sigma_{\hat{a}3} = 0.1327 & \hat{b}_3 = 0.5300 & \sigma_{\hat{b}3} = 0.1372 \\
\hat{a}_4 = 0.5160 & \sigma_{\hat{a}4} = 0.1610 & \hat{b}_4 = 0.4647 & \sigma_{\hat{b}4} = 0.1535 \\
\hat{a}_5 = -0.9941 & \sigma_{\hat{a}5} = 0.1297 & \hat{b}_5 = -0.3358 & \sigma_{\hat{b}5} = 0.1248 \\
\hat{a}_6 = 0.7828 & \sigma_{\hat{a}6} = 0.1694 & & \\
\hat{a}_7 = -0.2935 & \sigma_{\hat{a}7} = 0.1170 & & \\
\hat{a}_8 = 0.6018 & \sigma_{\hat{a}8} = 0.0995 & & \\
\hat{a}_9 = -0.0802 & \sigma_{\hat{a}9} = 0.0926 & & \\
\hat{a}_{10} = 0.0601 & \sigma_{\hat{a}10} = 0.0759 & & \\
\hat{a}_{11} = -0.1982 & \sigma_{\hat{a}11} = 0.0460 & &
\end{array}
\tag{9}
$$

and the final model is given as

Table 10: Estimation Results

$na = 11$, $nb = 5$	
Sum Square Error	3.5292
σ_e^2	0.0036
α	23.6848
Q	138.0579

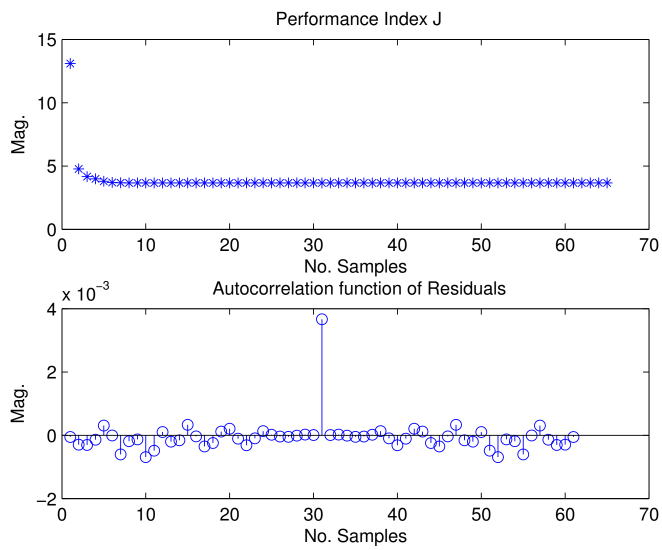

Fig. 9: Performance Index and A.C.F of Residual[$na = 7, nb = 5$]

6 FREQUENCY RESPONSE ANALYSIS

Frequency response and Impulse response analysis are the tools to evaluate the behavior of systems. Two linear systems are identical if they have the same frequency response or impulse response. Due to this fact we will do frequency response analysis in order to chose the best (simplest) models derived in section 5.3 and 5.4. Figure(11) and (12) shows the magnitude response and phase response of the two filters derived in section 5 (ARMA(7,5) and ARMA(11,5)). From the figures it is clear that they have almost the same frequency responses within the given frequency range. Since they both have almost the same frequency responses in the given frequency range, their sum square errors are close to each other, the first row of their GPAC residuals are small (≈ 0) and the variance of their residuals are small so their behavior are very close to each other and it is better to choose the model with the lower order. Hence the best model which represent these set of data is ***ARMA(7,5)***.

7 FUTURE WORK

The model derived in this paper is a linear model with delayed inputs and outputs. The future work of this paper is to deploy a nonlinear model for more accurate prediction. The nonlinear autoregressive moving average (NARMA) is the next step for this research work. The Levenberg-Marduard algorithm is a nonlinear optimization method which can be used for the parameter estimation of NARMA model.

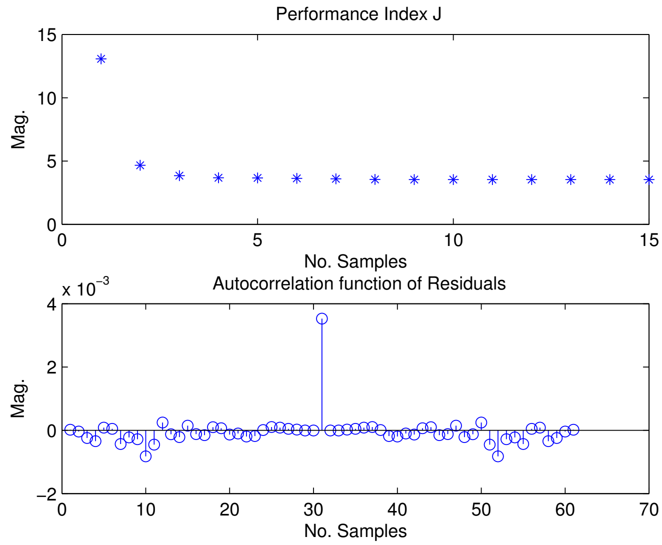

Fig. 10: Performance Index and A.C.F of Residual[$na = 11, nb = 5$]

8 SUMMARY AND CONCLUSIONS

In this project we tried to find out the simplest possible linear model for a real set of data. We used an Auto Regressive Moving Average (ARMA) model to represent the set of given data. Leuenbrg-Marqurrdt Algorithm implemented to estimate the model parameters. The statistic Chi-Square test (Diagnostic test) used to evaluate the whiteness of residual. Diagnostic test shows the time to stop increasing the model order. As we mentioned in the body of this report (section 5) none of the model which we derived could pass the Chi-Square test. In fact this is not really surprising because we are trying to fit a nonlinear data into a linear model. Although we can get close to the ideal linear model which represent the data but expecting an ARMA model to fully represent a nonlinear data is not realistic.

One of the *limitation* which we faced in this project was the inherent limitation of ARMA model. According to our observation no ARMA model can fully represent this set of data. One possible solution for better performance could be changing the model and try other model such as ARMAX or Box-Jenkins model (for detail of these models refer to [7]). If changing the model in not possible then we *suggest* to chose the other part of data which is more linear. Usually the initial part of speech signals (consonant letter) could be modeled by linear models (Figure 3). There might be a chance of better performance by choosing the other part of data.

Table 11: GPAC of Residual [$na = 11$, $nb = 5$]

	k = 1	k = 2	k = 3	k = 4	k = 5	k = 6	k = 7	k = 8	k = 9	k = 10
j=0	-0.0020	-0.0014	0.0055	0.0129	0.0238	0.0291	0.0016	-0.0528	-0.0573	-0.0311
j=1	0.7077	-0.0092	0.0088	0.0026	0.0080	0.0279	0.9824	-0.0545	-0.0288	0.0422
j=2	-3.9405	-3.7523	0.0072	-0.0190	0.0020	0.0155	0.0160	-0.0378	-0.0686	-0.0458
j=3	2.3141	-0.6817	-1.8025	-0.0127	0.1112	0.0145	0.0520	-0.0467	-0.1036	0.2047
j=4	1.8466	-5.6889	-0.5996	-5.2828	0.0214	-0.0093	0.0225	-0.0196	-0.0566	-0.0418
j=5	1.2241	-4.2593	32.8473	-4.2699	-1.8216	0.0761	0.0180	-0.0573	-0.0500	0.1050
j=6	0.0511	-1.8870	1.9630	-6.9566	-17.1595	4.0386	0.0672	-0.0272	-0.0640	-0.0378
j=7	-35.3286	-1.8574	-4.3244	-3.9099	-3.3738	-10.8300	-4.4085	-0.1104	-0.0685	0.0705
j=8	1.0658	-0.5666	1.0544	-2.2757	6.7096	-5.7740	13.6584	-8.4852	-0.0605	-0.0855
j=9	0.5182	0.8119	-0.4666	-0.9763	0.8566	1.6358	-1.0422	-1.0979	1.3314	0.04

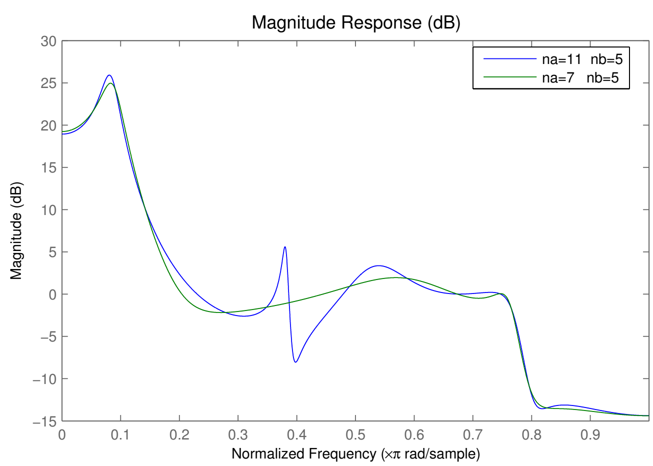

Fig. 11: Frequency Response Analysis(Magnitude Response)

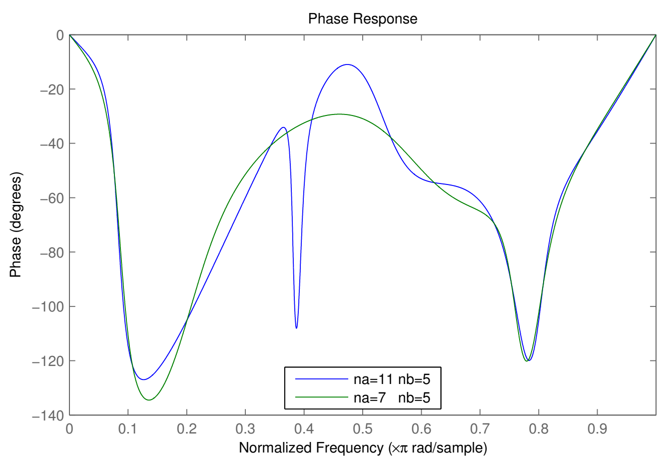

Fig. 12: Frequency Response Analysis(Phase Response)

References

1. George Box and Jenkins. *Time series analysis: Forecasting and Control*. Holden-Day, San Francisco, 1970.
2. Peter J. Brckwell and Richard A. Davis. *Theory and Methods*. Springer-Verlag, 1991.
3. Koopman S.J Commandeur J.J.F. *Introduction to State Space Time Series Analysis*. Oxford University Press, 2007.
4. Dr. Martin Hagan. *My Estimation Theory (ECEN 5523) Class Note*. Oklahoma State University, May 2007.
5. Reza Jafari. *Stability Analysis of Recurrent Neural-Based Controllers*. Phd dissertation, Oklahoma State University, May 2012.
6. Kenneth Levenberg. A method for the solution of certain non-linear problems in least squares. *Quarterly of Applied Mathematics 2*, pages 164–168, 1944.
7. Lennart Ljung. *System Identification, Theory for user*. Prentice Hall Information and System Sciences Series, New Jersy, 1987.
8. Donald Marquardt. An algorithm for least-squares estimation of nonlinear parameters. *SIAM Journal on Applied Mathematics*, pages 431–441, 1963.
9. Jerry M. Mendel. *Lessons in Estimation Theory for Signal Processing, Communication and Control*. PRENTICE HALL, New Jersey, 1995.
10. Jorge Nocedal. *Numerical Optimization,*. Springer, 2nd edition, 2006.
11. Pankratz. *Forecasting with univariate Box-Jenkins models: concepts and cases*. John Wiley and Sons, New York, 1983.
12. M. Hagan R. Jafari. Global stability analysis using the method of reduction of dissipativity domain. *IEEE*, pages 2550–2556, July 2011.
13. R. Dhaouadi R. Jafari. *Adaptive PID Control of Nonlinear Servomechanism Using Recurrent Neural Networks*, chapter 15, pages 275–296. InTech, Jan. 2011.