1. **ibm data:**

   The random walk model of first differences is chosen to be the suggest model of ibm data. That is

   \[(1 - B)Y_t = e_t\]

   where \(e_t\) is a mean zero white noise process.

   For the diagnose part, first we do runs test to test if first differences are independent. A runs test on the first differences procedures the following output:

   ```
   > runs(diff(ibm))#Run test.
   $pvalue
   [1] 0.258
   $observed.runs
   [1] 172
   $expected.runs
   [1] 183.2391
   ```

   The \(p\)-value= 0.258 for the test is not small, so we would not reject the null hypothesis. This test does not provide evidence of dependency of the first differences. However, the Shapiro-Wilk test suggest the non-normality of the first difference.

   Second, we do Shapiro-Wilk test to see if the first differences are normal distributed. The Shapiro-Wilk test on the first differences procedures the output:

   ```
   > shapiro.test(diff(ibm))
   Shapiro-Wilk normality test
data: diff(ibm)
W = 0.9585, p-value = 1.094e-08
   ```

   The \(p\)-value= 1.094 \times 10^{-8} is extremely small, so we would reject null hypothesis. The test provide strong evidence of non-normality for the first differences. Please see Figure 1. The shape of histogram of first differences is bell-shaped with heavy tails. So \(t\)-distribution could be a possible choice. The qq-plot of \(t\)-distribution with 5 degree of freedom looks good. The followings are R commands to make the histogram and qq-plot.
Figure 1: Left: Histogram of first differences of \texttt{ibm} data. Right: QQ plot for \texttt{diff(ibm)}

```r
par(mfrow=c(1,2))
hist(diff(ibm))
qqplot(qt(ppoints(diff(ibm)),df = 7.6), diff(ibm), xlab = "Q-Q plot for t(5)"
qqline(diff(ibm))
```

The \texttt{R} function \texttt{tsdiag} produces the plot in Figure 2. The top plot displays the residuals plotted through time (without connecting lines). The middle plot displays the sample ACF of the residuals. The bottom plot displays the \textit{p}–values of the modified Ljung-Box test for various values of \textit{K}. A horizontal line at $\alpha = 0.05$ is added. For the \texttt{ibm} data, we see in Figure 2 that all of the modified Ljung-Box test \textit{p}–value are mostly larger than 0.05 at early \textit{K}. However, they are close to $\alpha = 0.05$ and turn to significant at the end. It raises my concerns that this model may not be adequate. The autocorrelations are looking good even thought there is significant large value at long lags.

We now overfit using an ARIMA(1,1,0) and an ARIMA(0,1,1) model. Here is the \texttt{R} output from all three model fits:

```r
> arima(ibm,order=c(0,1,0),method='ML')
sigma^2 estimated as 52.62: log likelihood = -1251.37, aic = 2502.74
> arima(ibm,order=c(1,1,0),method='ML')
Coefficients:
  ar1
0.0869
```
Figure 2: *ibm* data. Residual graphics and modified Ljung-Box $p-$values for ARIMA(0,1,0) fit. This figure was created using the `tsdiag` function in R.
In the ARIMA(1,1,0) overfit, we see that a 95 percent confidence interval for \( \phi \), the additional AR model parameter, is

\[
0.0869 \pm 1.96(0.0519) \implies (-0.189, 0.015),
\]

which does include 0. Therefore, \( \hat{\phi} \) is not statistically different than zero, which suggests that the ARIMA(1,1,0) model is not necessary. In the ARIMA(0,1,1) overfit, we see that a 95 percent confidence interval for \( \phi \), the additional AR model parameter, is

\[
0.0864 \pm 1.96(0.0512) \implies (-0.187, 0.014),
\]

which does include 0. Therefore, \( \hat{\theta} \) is not statistically different than zero, which suggests that the ARIMA(0,1,1) model is not necessary. The following table summarizes the output. Because the additional estimates in the overfit models are not statistically different from zero, there is no reason to further consider either model. Note also how the estimate of \( \theta \) and \( \phi \) becomes less precise in the two larger models.

So we still conclude that the final model is

\[
(1 - B)Y_t = e_t
\]

where \( e_t \) is a mean zero white noise process with \( e_t \) following \( t \)-distribution with 5 degrees of freedom.
internet data:
The ARIMA(2,2,0) model of first differences is chosen to be the suggest model of internet data. That is
\[(1 - \phi_1 B - \phi_2 B^2)(1 - B)^2 Y_t = e_t.\]

Here is the R output from fitting an ARIMA(2,2,0) model via maximum likelihood:

```R
> arima(internet, order=c(2,2,0), method='ML')

Coefficients:
            ar1      ar2
          0.2579 -0.4407
                  s.e.    0.0915   0.0906
sigma^2 estimated as 10.13: log likelihood = -252.73, aic = 509.46
```

Therefore, the fitted ARIMA(2,2,0) model is
\[\nabla^2 Y_t = 0.2579 \nabla^2 Y_{t-1} - 0.4407 \nabla^2 Y_{t-2} + e_t.\]

The white noise variance estimate, using CLS, is \(\hat{\sigma}_e^2 \approx 10.13\). Figure 3 displays the time series plot (upper right), the histogram (lower left), and the qq plot (lower right) of the standardized residuals. The histogram and the qq plot show no gross departures from normality. The time series plot of the standardized residuals displays no noticeable patterns and looks like a stationary random process.

The R output for the Shapiro-Wilk and runs tests is given below:

```R
> shapiro.test(rstandard(internet.arima.fit))
    Shapiro-Wilk normality test
data: rstandard(internet.arima.fit)
    W = 0.9896, p-value = 0.6305
> runs.test(rstandard(internet.arima.fit))
Error in runs.test(rstandard(internet.arima.fit)) : x is not a factor
> runs(rstandard(internet.arima.fit))
$pvalue
[1] 0.999
$observed.runs
[1] 50
$expected.runs
[1] 50.5
```

The Shapiro-Wilk test does not reject normality (\(p\text{-value}=0.6305\)). The runs test does not reject independence (\(p\text{-value}=0.999\)). For the internet data, (standardized) residuals from a ARIMA(2,2,0) fit look to reasonably
Figure 3: internet data. Upper left: internet time series. Upper right: Standardized residuals from an ARIMA(2,2,0) fit with zero line added. Lower left: Histogram of the standardized residuals from ARIMA(2,2,0) fit. Lower right: QQ plot of the standardized residuals from ARIMA(2,2,0) fit.
Figure 4: internet data. Residual graphics and modified Ljung-Box $p$-values for ARIMA(2,2,0) fit. This figure was created using the `tsdiag` function in R.
satisfy the normality and independence assumptions. The R function `tsdiag` produces the plot in Figure 4. The top plot displays the residuals plotted through time (without connecting lines). The middle plot displays the sample ACF of the residuals. The bottom plot displays the $p$-values of the modified Ljung-Box test for various values of $K$. A horizontal line at $\alpha = 0.05$ is added. For the `internet` data, we see in Figure 4 that all of the modified Ljung-Box test $p$-value are larger than 0.05, lending further support of the ARIMA(2,2,0) model. In other words, the residual output in Figure 4 fully supports the ARIMA(2,2,0) model.

Our residual analysis suggests that an ARIMA(2,2,0) model for the `internet` data is reasonable. We now overfit using an ARIMA(3,2,0) and an ARIMA(2,2,1) model. Here is the R output from all three model fits:

```r
> arima(internet, order=c(2,2,0), method='ML')
Coefficients:
       ar1  ar2
 0.2579 -0.4407
s.e. 0.0915 0.0906
sigma^2 estimated as 10.13: log likelihood = -252.73, aic = 509.46
> arima(internet, order=c(2,2,1), method='ML') #overfitting diagnostics
Coefficients:
       ar1  ar2  ma1
 0.3512 -0.4572 -0.1161
s.e. 0.2189 0.0937 0.2502
sigma^2 estimated as 10.10: log likelihood = -252.63, aic = 511.26
> arima(internet, order=c(3,2,0), method='ML') #overfitting diagnostics
Coefficients:
       ar1  ar2  ar3
 0.2390 -0.4298 -0.0430
s.e. 0.1021 0.0943 0.1033
sigma^2 estimated as 10.11: log likelihood = -252.65, aic = 511.29
```

In the ARIMA(2,2,1) overfit, we see that a 95 percent confidence interval for $\phi$, the additional MA model parameter, is

$$-0.1161 \pm 1.96(0.2502) \implies (-0.606, 0.374),$$

which does include 0. Therefore, $\hat{\theta}$ is not statistically different than zero, which suggests that the ARIMA(2,2,1) model is not necessary. In the ARIMA(3,2,0) overfit, we see that a 95 percent confidence interval for $\phi_3$, the additional AR model parameter, is

$$-0.0430 \pm 1.96(0.1033) \implies (-0.245, 0.159),$$
which does include 0. Therefore, \( \hat{\phi}_3 \) is not statistically different than zero, which suggests that the ARIMA(3,2,0) model is not necessary. The following table summarizes the output. Because the additional estimates in the overfit

<table>
<thead>
<tr>
<th>Model</th>
<th>( \theta(\hat{\theta}) )</th>
<th>Additional estimate</th>
<th>Significant?</th>
<th>( \hat{\sigma}^2 )</th>
<th>AIC</th>
</tr>
</thead>
<tbody>
<tr>
<td>ARIMA(2,2,0)</td>
<td>0.2579(0.0915)</td>
<td>-</td>
<td>-</td>
<td>10.13</td>
<td>509.46</td>
</tr>
<tr>
<td></td>
<td>-0.4407(0.0906)</td>
<td>-</td>
<td>-</td>
<td>10.13</td>
<td>509.46</td>
</tr>
<tr>
<td>ARIMA(2,2,1)</td>
<td>-0.1161(0.2501)</td>
<td>( \hat{\theta} )</td>
<td>no</td>
<td>10.10</td>
<td>511.26</td>
</tr>
<tr>
<td>ARIMA(3,2,0)</td>
<td>-0.0430(0.1033)</td>
<td>( \hat{\phi}_3 )</td>
<td>no</td>
<td>10.11</td>
<td>511.29</td>
</tr>
</tbody>
</table>

models are not statistically different from zero, there is no reason to further consider either model. Note also how the estimate of \( \theta \) and \( \phi \) becomes less precise in the two larger models.

After all, ARIAM(2,2,0) model performs well and I feel satisfied to choose this model.
gasprices data:
The ARIMA(0,1,1) model of first differences is chosen to be the suggest model of gasprices data. That is

\[ ((1 - B)Y_t) = (1 - \theta B)e_t. \]

Here is the R output from fitting an ARIMA(0,1,1) model via maximum likelihood:

```r
> gasprices.arima.fit
Coefficients:
   ma1
  0.4418
s.e.  0.0625
sigma^2 estimated as 0.002408: log likelihood = 229.65, aic = -457.3
```

Therefore, the fitted ARIMA(0,1,1) model is

\[ \nabla Y_t = 0.4418e_{t-1} + e_t. \]

The white noise variance estimate, using CLS, is \( \hat{\sigma}_e^2 \approx 0.002408 \). Figure 5 displays the time series plot (upper right), the histogram (lower left), and the qq plot (lower right) of the standardized residuals. The histogram show the distribution of standardized residuals is a little right-skewed. The qq plot show a little off normal assumption. Time series plot of the standardized residuals displays non-constant variance. The R output for the Shapiro-Wilk and runs tests is given below:

```r
> shapiro.test(rstandard(gasprices.arima.fit))
Shapiro-Wilk normality test
data:  rstandard(gasprices.arima.fit)
W = 0.9777, p-value = 0.01824
> runs(rstandard(gasprices.arima.fit))
$pvalue
[1] 0.759
$observed.runs
[1] 71
$expected.runs
[1] 73.33103
```

The Shapiro-Wilk test does reject normality (\( p-value=0.01824 \)). However, the runs test does not reject independence (\( p-value= 0.759 \)). For the ibm data, (standardized) residuals from a ARIMA(0,1,1) fit satisfy the independence assumptions but not normality. The R function tsdiag produces the plot in Figure 6. The top plot displays the residuals plotted through time (without
Figure 5: gasprices data. Upper left: gasprices time series. Upper right: Standardized residuals from an ARIMA(0,1,1) fit with zero line added. Lower left: Histogram of the standardized residuals from ARIMA(0,1,1) fit. Lower right: QQ plot of the standardized residuals from ARIMA(0,1,1) fit.
Figure 6: gasprices data. Residual graphics and modified Ljung-Box $p$-values for ARIMA$(0,1,1)$ fit. This figure was created using the tsdiag function in R.
connecting lines). The middle plot displays the sample ACF of the residuals. The bottom plot displays the $p$–values of the modified Ljung-Box test for various values of $K$. A horizontal line at $\alpha = 0.05$ is added. For the gasprices data, we see in Figure 6 that all of the modified Ljung-Box test $p$–value are larger than 0.05, lending further support of the ARIMA(0,1,1) model. In other words, the residual output in Figure 6 fully supports the ARIMA(0,1,1) model.

Our residual analysis suggests that an ARIMA(0,1,1) model for the ibm data is reasonable. We now overfit using an ARIMA(1,1,1) and an ARIMA(0,1,2) model. Here is the R output from all three model fits:

```r
> arima(gasprices,order=c(0,1,1),method='ML')
Coefficients:
   ma1
0.4418
s.e. 0.0625
sigma^2 estimated as 0.002408: log likelihood = 229.65, aic = -457.3
```

```r
> arima(gasprices,order=c(1,1,1),method='ML')
Coefficients:
   ar1   ma1
0.3585 0.1713
s.e. 0.1451 0.1479
sigma^2 estimated as 0.002324: log likelihood = 232.18, aic = -460.35
```

```r
> arima(gasprices,order=c(0,1,2),method='ML')
Coefficients:
   ma1   ma2
0.5433 0.1946
s.e. 0.0843 0.0779
sigma^2 estimated as 0.002312: log likelihood = 232.53, aic = -461.06
```

In the ARIMA(1,1,1) overfit, we see that a 95 percent confidence interval for $\phi$, the additional AR model parameter, is

$$-0.3585 \pm 1.96(0.1451) \implies (-0.643, -0.074),$$

which does not include 0. Therefore, $\hat{\phi}$ is statistically different than zero, which suggests that the ARIMA(1,1,1) model is worth considered. In the ARIMA(0,1,2) overfit, we see that a 95 percent confidence interval for $\theta_2$, the additional AR model parameter, is

$$0.1946 \pm 1.96(0.0779) \implies (0.042, 0.347),$$

which does not include 0. Therefore, $\hat{\theta}_2$ is statistically different than zero, which suggests that the ARIMA(0,1,2) model is worth considered. The following table summarizes the output. Because the additional estimates in the
Table 3: Overfit ARIMA(0,1,1) model

<table>
<thead>
<tr>
<th>Model</th>
<th>$\hat{\theta}(\hat{se})$</th>
<th>Additional estimate</th>
<th>Significant?</th>
<th>$\hat{\sigma}^2$</th>
<th>AIC</th>
</tr>
</thead>
<tbody>
<tr>
<td>ARIMA(0,1,1)</td>
<td>0.2579(0.0915)</td>
<td>–</td>
<td>–</td>
<td>0.002408</td>
<td>-457.3</td>
</tr>
<tr>
<td>ARIMA(1,1,1)</td>
<td>0.3585(0.1451)</td>
<td>$\hat{\phi}$</td>
<td>yes</td>
<td>0.002324</td>
<td>-460.35</td>
</tr>
<tr>
<td>ARIMA(0,1,2)</td>
<td>0.1946(0.0779)</td>
<td>$\hat{\theta}_2$</td>
<td>yes</td>
<td>0.002312</td>
<td>-461.06</td>
</tr>
</tbody>
</table>

overfit models are statistically different from zero, considering further additional models is worth.

After all, the original model is not totally satisfied after diagnosis. It is needed to do further search to find more appropriate model in the future.
2.  
2.(a) Because the numbers of home runs are “counts”, this suggests that a transformation is needed. The Box-Cox transformation output in Figure 7 shows that $\lambda = 0.5$ resides in an approximate 95 percent confidence interval for $\lambda$. Recall that $\lambda = 0.5$ corresponds to the square-root transformation. Taking first differences of the squared-root of data is need because there should exist trend of the square-root of home runs data.

2.(b) The ACF of $\{Z_t\}$ should decay very slowly because there is trend and it is not a stationary process. Comparing with ACF of $\{Z_t\}$, we expected to see ACF of $\{\nabla Z_t\}$ decay to zero faster.

2.(c) According to the output from R, we have $\hat{\theta} = -0.2488$ and estimated standard deviation 0.1072. An approximate 95 percent confidence interval for $\theta$ is

$$-0.2488 \pm 1.96(0.1072) \implies (-0.459, -0.039).$$

We are 95 percent confident that $\theta$ is between $-0.459$ and $-0.039$.

2.(d) According to Shapiro test and runs test of standardized residuals, we do not have evidence to reject that the residuals are mean zero white process. However, the modified Ljung-Box test $p$-values raise serious concerns over adequacy of the IMA(1,1) model fit.
3.

Table 4: Standardized Residuals after IMA(1,1) model fitted

<table>
<thead>
<tr>
<th>Time lag</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACF</td>
<td>0.882</td>
<td>0.811</td>
<td>0.733</td>
<td>0.690</td>
<td>0.674</td>
<td>0.668</td>
<td>0.661</td>
<td>0.621</td>
<td>0.568</td>
<td>0.519</td>
</tr>
<tr>
<td>PACF</td>
<td>0.882</td>
<td>0.144</td>
<td>-0.029</td>
<td>0.111</td>
<td>0.159</td>
<td>0.093</td>
<td>0.050</td>
<td>-0.097</td>
<td>-0.087</td>
<td>-0.015</td>
</tr>
</tbody>
</table>

Table 5: Ljung-Box Test and Fisher and Gallagher Test

<table>
<thead>
<tr>
<th>m</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>$Q_r$</td>
<td>0.156</td>
<td>4.539</td>
<td>10.917</td>
<td>13.624</td>
<td>13.941</td>
<td>15.364</td>
<td>17.934</td>
<td>17.984</td>
<td>18.031</td>
</tr>
<tr>
<td>p-value</td>
<td>0.693</td>
<td>0.103</td>
<td>0.012</td>
<td>0.009</td>
<td>0.016</td>
<td>0.018</td>
<td>0.012</td>
<td>0.018</td>
<td>0.035</td>
</tr>
<tr>
<td>$Q_W$</td>
<td>0.079</td>
<td>1.566</td>
<td>3.903</td>
<td>5.847</td>
<td>7.196</td>
<td>8.363</td>
<td>9.559</td>
<td>10.495</td>
<td>11.249</td>
</tr>
<tr>
<td>p-value</td>
<td>1.000</td>
<td>0.605</td>
<td>0.140</td>
<td>0.053</td>
<td>0.036</td>
<td>0.029</td>
<td>0.023</td>
<td>0.021</td>
<td>0.022</td>
</tr>
<tr>
<td>$M_W$</td>
<td>0.079</td>
<td>1.566</td>
<td>4.068</td>
<td>6.206</td>
<td>7.849</td>
<td>9.023</td>
<td>9.926</td>
<td>10.726</td>
<td>11.437</td>
</tr>
<tr>
<td>p-value</td>
<td>1.000</td>
<td>0.605</td>
<td>0.121</td>
<td>0.039</td>
<td>0.022</td>
<td>0.018</td>
<td>0.018</td>
<td>0.018</td>
<td>0.020</td>
</tr>
</tbody>
</table>

According to Table 5, Fisher and Gallagher test suggests that IMA(1,1) may not be adequate just like Ljung-Box test does.

The followings are R commands to calculate ACF, PACF, $Q_r$, $Q_W$, and $M_W$.

```r
round(acf(homeruns,plot=F,lag.max=10)$acf,3) #ACF
round(pacf(homeruns,plot=F,lag.max=10)$acf,3) #PACF
Q_LjungBox = function(r,m){ #LjungBox test statistic
  n = length(r); mn = length(m)
  Q_LjungBox = array(,mn)
  for (i in 1:mn){
    k = 1:m[i]
    acf_r = acf(r,plot=F,lag.max=m[i])$acf
    Q_LjungBox[i] = n*(n+2)*sum((acf_r^2)/(n-k))
  }
  return(Q_LjungBox)
}
p_value_for_Q_LjungBox = function(Q_LjungBox,m,p=0,q=1){
  mn = length(m); p_value = array(,mn)
  for (i in 1:mn){
    p_value[i] = 1-pchisq(Q_LjungBox[i],df=m[i]-p-q)
  }
  return(p_value)
}
```

16
\[ m = 2:10 \\
round(Q_LjungBox(rstandard(homerun.fit),m),3) \\
round(p_value_for_Q_LjungBox(Q_LjungBox(rstandard(homerun.fit),m),m),3) \\
QW = function(r,m){ 
    n = length(r); mn = length(m) 
    QW = array(,mn) 
    for (i in 1:mn){ 
        k = 1:m[i] 
        acf_r = acf(r,plot=F,lag.max=m[i])$acf 
        QW[i] = n*(n+2)*sum((m[i]-k+1)/m[i] *(acf_r^2)/(n-k)) 
    } 
    return(QW) 
} \\
MW = function(r,m){ 
    n = length(r); mn = length(m) 
    MW = array(,mn) 
    for (i in 1:mn){ 
        k = 1:m[i] 
        pacf_r = pacf(r,plot=F,lag.max=m[i])$acf 
        MW[i] = n*(n+2)*sum((m[i]-k+1)/m[i] *(pacf_r^2)/(n-k)) 
    } 
    return(MW) 
} \\
p_value_for_QM = function(QM, m){ 
    mn = length(m) 
    p_value = array(,mn) 
    for (i in 1:mn){ 
        alpha = (3/4)*(m[i]*(m[i]+1)^2)/(2*m[i]^2+3*m[i]+1-6*m[i]) #Equation(7) 
        beta = (2/3)*(2*m[i]^2+3*m[i]+1-6*m[i])/(m[i]*(m[i]+1)) #Equation(8) 
        p_value[i] = 1-pgamma(QM[i],shape=alpha,scale=beta) 
    } 
    return(p_value) 
} \\
m = 2:10 \\
round(QW(rstandard(homerun.fit),2:10),3) \\
round(p_value_for_QM(QW(rstandard(homerun.fit),m),m),3) \\
round(MW(rstandard(homerun.fit),2:10),3) \\
round(p_value_for_QM(MW(rstandard(homerun.fit),m),m),3)