CS286 Machine Learning

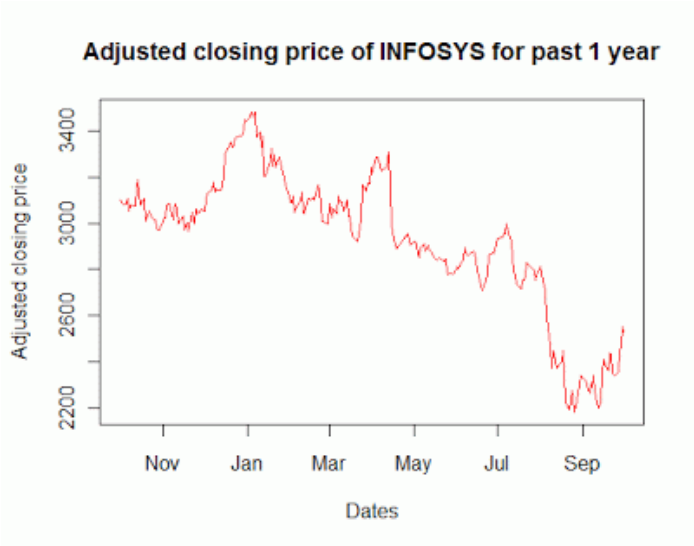
Class Project

5/5/2015

Git Repo: https://github.com/gyoho/stock-market-prediction

Why choose adjusted closing price

The closing price of a stock is exactly that: the price of that stock at the close of the trading day. The adjusted closing price uses the closing price as a starting point, but it takes into account factors such as dividends, stock splits and new stock offerings. The adjusted closing price represents a more accurate reflection of a stock's value, since distributions and new offerings can alter the closing price.



Important Characteristics to Consider First

- Is there a trend, meaning that, on average, the measurements tend to increase (or decrease) over time?
- Is there seasonality, meaning that there is a regularly repeating pattern of highs and lows related to calendar time such as seasons, quarters, months, days of the week, and so on?
- Are there outliers? In regression, outliers are far away from your line. With time series data, your outliers are far away from your other data.
- Is there a long-run cycle or period unrelated to seasonality factors?
- Is there constant variance over time, or is the variance non-constant?
- Are there any abrupt changes to either the level of the series or the variance?

Null hypothesis

Null hypothesis refers to a general statement or default position that there is no relationship between two measured phenomena. By disproving this, we can conclude there is a relationship. The alternative hypothesis is the one you would believe if the null hypothesis is concluded to be untrue.

Univariate time series

Stock price: Ordering is very important because there is dependency and changing the order could change the meaning of the data.

Stationary vs. Non-stationary

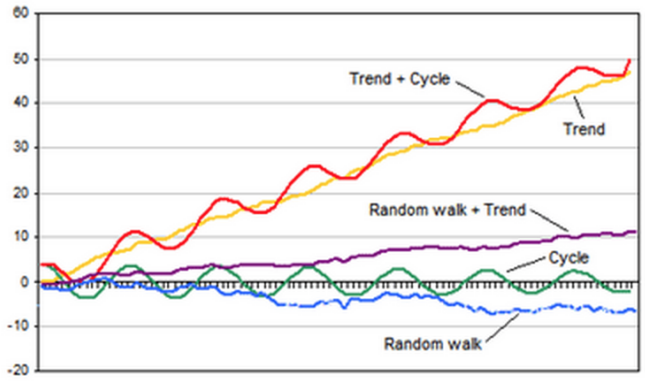
Stationary

- properties do not depend on the time
- constant mean and constant variance independent of time
- Ex) white noise

Non-stationary

- time series with trends, or with seasonality, are not stationary
- the trend and seasonality will affect the value of the time series at different times
- variable variance and a mean that does not remain near, or returns to a long-run mean over time

Table 1 Non-stationary behavior



Stock:

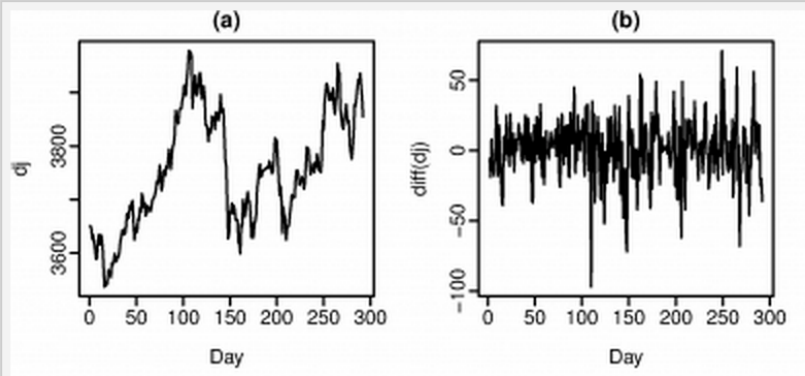
- **価格帯によって平均と分散は区間毎に違う**
- **変動の仕方によっても平均と分散は区間毎に違う**

Transform to Stationary

Motivation

Statistics doesn't work with non-stationary series. Since the mean and variance are not constant, we cannot use statistics make prediction.

Technique



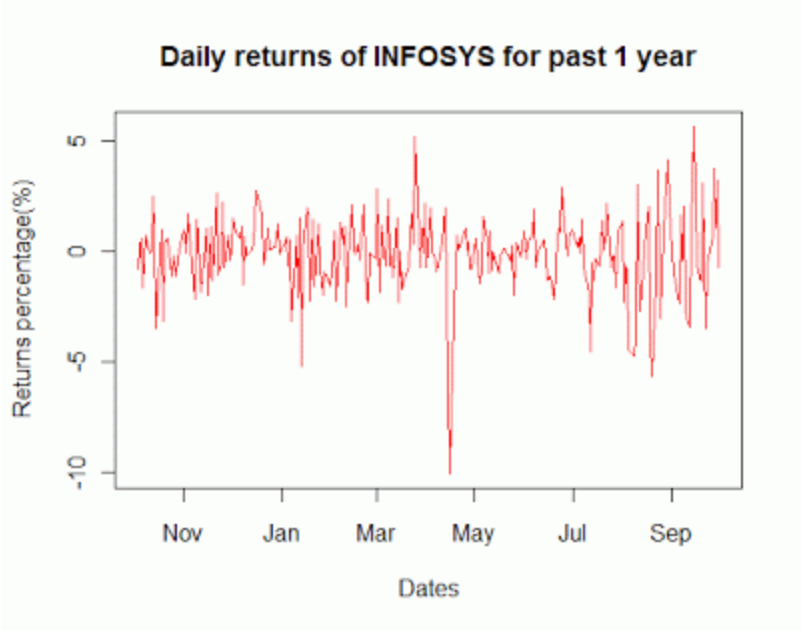
Notice how the Dow Jones index data was non-stationary in panel (a), but the daily changes were stationary in panel (b). This shows one way to make a time series stationary — compute the differences between consecutive observations. This is known as *differencing*. Transformations such as logarithms can help to stabilize the variance of a time series. Differencing can help stabilize the mean of a time series by removing changes in the level of a time series, and so eliminating trend and seasonality.

Differencing

- First-order differencing: $y'_t = y_t - y_t - 1$
- Second-order differencing: $y''_t = y'_t - y't - 1$

Remove the time and price dependency to make mean and variance constant against time. But, at the same time, we only get the relationship and lost the raw data. Thus, the result isn't guaranteed any more.

```
>> infy_ret <- 100*diff(log(infy[,2]))
```



The above plot the mean is fixed at 0 and the fluctuations are around that mean, that doesn't change with time

Autocorrelation

Def

Find the correlation between two different time series (original and lagged): lagged correlation.

Motivation

If you know a stock historically has a high positive autocorrelation value and you witnessed the stock making solid gains over the past several days, you might reasonably expect the movements over the upcoming several days (the leading time series) to match those of the lagging time series and to move upwards.

Model

- ARMA model: Autoregressive–moving-average model
- AR model: autoregressive (AR) model
  ※ AR model is a special case of the more general ARMA model of time series.

AR(*p*) model: p-order autoregressive model

p: the previous *p* terms and the noise term contribute to the output

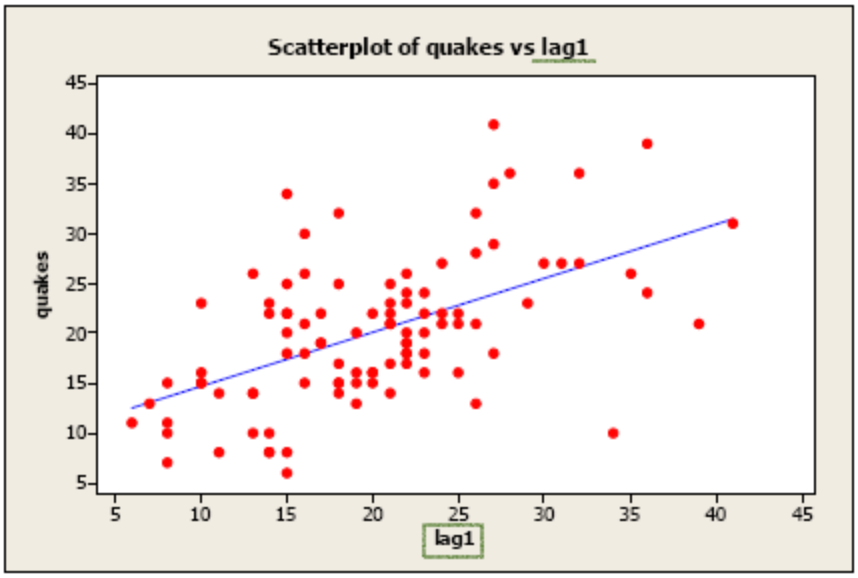
AR(1) model: The First-order Autoregression Model

Lag: lagged by x time units i.e) the value of xth past time
ex) Let x_t denote the value of the series at any particular time t, so x_{t-1} denotes the value of the series one time before time t. That is, x_{t-1} is the lag 1 value of x_t.

t	x_t	x_{t-1} (lag 1 value)
1	13	*
2	14	13
3	8	14
4	10	8
5	16	10

To see if there is a correlation x_t vs. x_{t-1}

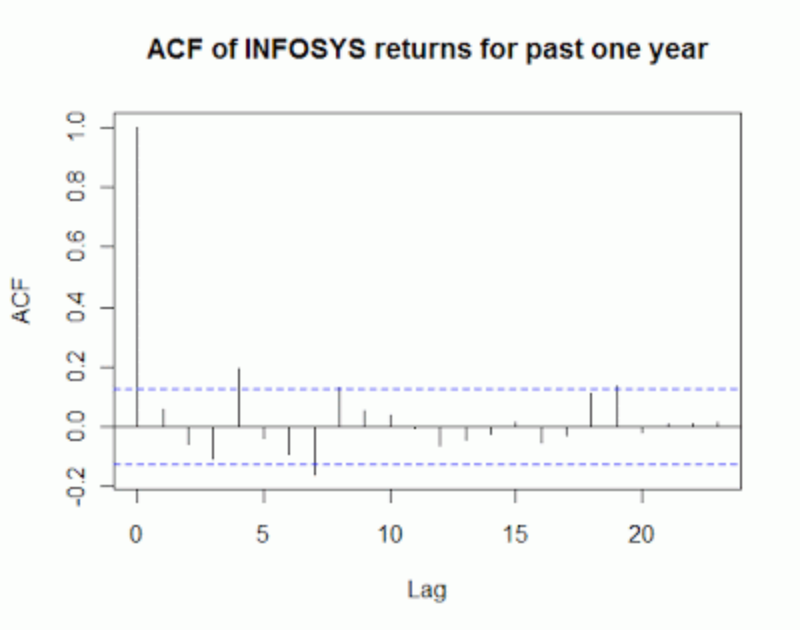
$x_t=\delta+\phi_1 x_{t-1}+w_t$



AutoCorrelation in R

calculate the autocorrelation up to lag=max

```
>> acf(x, lag.max, type)
      x: a univariate time series object
      lag.max: Default=10*log10(N/m)
      type: correlation(default), covariance, partial
```



Correlation Coefficient at different lags of the series

The blue dotted line is the 95% confidence interval.
confidence interval: precision and uncertainty associated with a particular sampling method.

Pearson product-moment correlation coefficient:

Measures the strength of the linear association between variables.

- The value of a correlation coefficient ranges between -1 and 1.
- The greater the absolute value of a correlation coefficient, the stronger the *linear* relationship.
- The strongest linear relationship is indicated by a correlation coefficient of -1 or 1.
- The weakest linear relationship is indicated by a correlation coefficient equal to 0.
- A positive correlation means that if one variable gets bigger, the other variable tends to get bigger.
- A negative correlation means that if one variable gets bigger, the other variable tends to get smaller.

Keep in mind that the Pearson correlation coefficient only measures linear relationships. Therefore, a correlation of 0 does not mean zero relationship between two variables; rather, it means zero linear relationship.

Linear Models in R

Describes the variable response by the variable term(s) i.e) x= term, y= response

```
>> lm(formula)
      formula: a symbolic description of the model to be fitted
      = response ~ terms
            response: response vector
            terms: a series of terms
                  first + second: all the terms in first together with all the
                        terms in second with duplicates removed
```

Regressing the returns from 1st to 7th lag

```
>> infy_out <- lm(infy_ret[8:length(infy_ret)] ~
infy_ret[8:length(infy_ret) - 1] + infy_ret[8:length(infy_ret) - 2] + infy_ret[8:length(infy_ret) - 3] +
infy_ret[8:length(infy_ret) - 4] + infy_ret[8:length(infy_ret) - 5] + infy_ret[8:length(infy_ret) - 6] +
infy_ret[8:length(infy_ret) - 7] )

>> summary(infy_out)
Residuals:
   Min     1Q     Median     3Q    Max
-4.6342 -0.7922 -0.0361  0.8504  3.5141
```

- Quick summary of the distribution:

It should be roughly symmetrical about mean, the median should be close to 0, the 1Q and 3Q values should ideally be roughly of similar absolute magnitude etc.

Coefficients:

p-Value

	Estimate	Std. Error	t value	Pr(>|t|)	
(Intercept)	-0.09316	0.11321	-0.823	0.41140	
infy_ret[8:length(infy_ret) - 1]	0.08158	0.06479	1.259	0.20920	
infy_ret[8:length(infy_ret) - 2]	-0.04017	0.06537	-0.614	0.53950	
infy_ret[8:length(infy_ret) - 3]	-0.10049	0.06528	-1.539	0.12504	
infy_ret[8:length(infy_ret) - 4]	0.20153	0.06457	3.121	0.00203	**
infy_ret[8:length(infy_ret) - 5]	-0.08566	0.06568	-1.304	0.19344	
infy_ret[8:length(infy_ret) - 6]	-0.06849	0.06584	-1.040	0.29928	
infy_ret[8:length(infy_ret) - 7]	-0.12395	0.06621	-1.872	0.06241	.

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Multiple R-squared: 0.08717, Adjusted R-squared: 0.05998

- The coefficient of the 4th lag is statistically significant
- p=0.02: significant relationship in the linear regression model in 4th lag
- ~ 6% of the explanation is provided by the above regression

y = ax + b;

a: slope	infy_ret[...]	0.20153
b: y-intercept	intercept	-0.09316

p-Value

Evaluate the hypothesis test. The value indicates there is a probability of 100*p% that you will mistakenly reject the hypothesis. The smaller, the stronger disprove the null hypothesis.
The p-value is a number between 0 and 1 and interpreted in the following way:

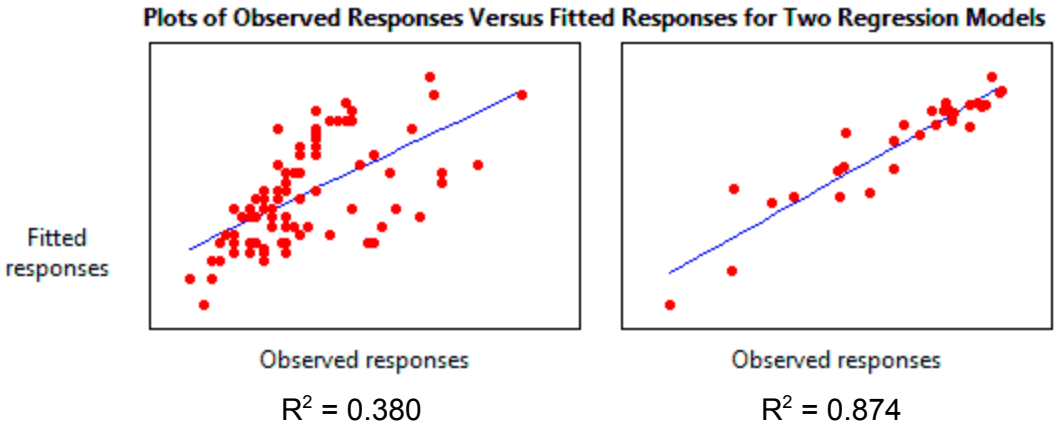
- A small p-value (typically ≤ 0.05) indicates strong evidence against the null hypothesis, so you reject the null hypothesis.
- A large p-value (> 0.05) indicates weak evidence against the null hypothesis, so you fail to reject the null hypothesis.
- p-values very close to the cutoff (0.05) are considered to be marginal (could go either way). Always report the p-value so your readers can draw their own conclusions.

R^2: coefficient of determination

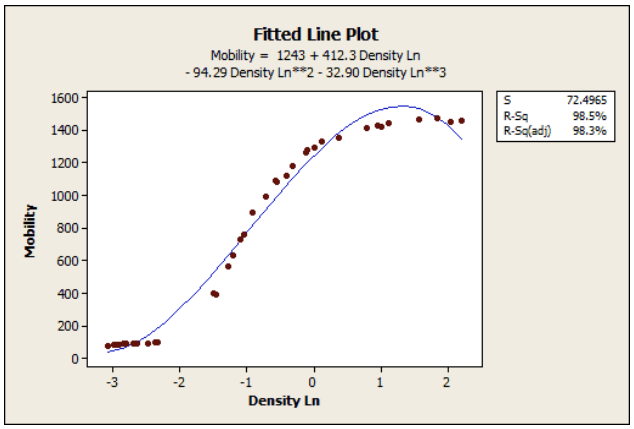
Measure of how close the data are to the fitted regression line.

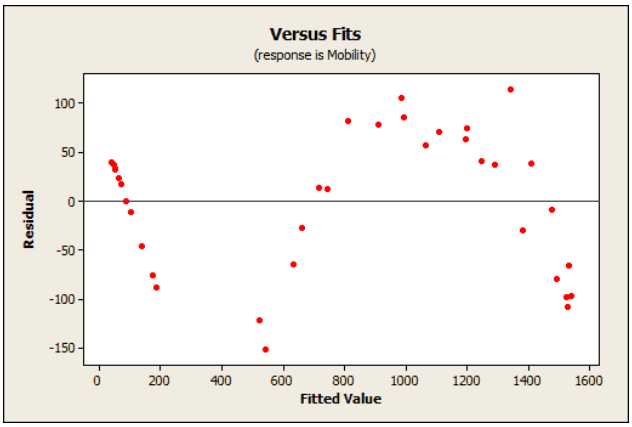
R-squared = Explained variation / Total variation
R-squared is always between 0 and 100%: higher R^2, model fits better

- 0% indicates that the model explains none of the variability of the response data around its mean.
- 100% indicates that the model explains all the variability of the response data around its mean.



R^2 = 0.380 R^2 = 0.874





The fitted line plot shows that these data follow a nice tight function and the R-squared is 98.5%, which sounds great. However, look closer to see how the regression line systematically over and under-predicts the data (bias) at different points along the curve. You can also see patterns in the Residuals versus Fits plot, rather than the randomness that you want to see. This indicates a bad fit, and serves as a reminder as to why you should always check the residual plots.

R^2 is necessary But not sufficient

ARMA Model

Autoregressive model + Moving-average model

Understanding:

The current value of the time series z_t will depend on the past value of the series z_{t-1} and will correct itself to the error made in the last time period ε_{t-1}.

ARMA(1):

$$z_t = \alpha + \phi z_{t-1} + \theta \epsilon_{t-1} + \epsilon_t$$

Autoregressive(1):

$$x_t = \delta + \phi_1 x_{t-1} + w_t$$

Moving-average(1):

$$x_t = \mu + w_t + \theta_1 w_{t-1}$$

ARIMAX: x_t=exogenous variable

$$z_t = \alpha + \phi z_{t-1} + \theta \epsilon_{t-1} + \gamma x_t + \epsilon_t$$

R syntax

```
>> arma(time-series_data, order = c(0L, 0L))
      c: order of AR and MA
ex) order=c(2,2)
```

Coefficient(s):

	Estimate	Std. Error	t value	Pr(>|t|)
ar1	1.563452	0.110834	14.106	< 2e-16 ***
ar2	-0.700315	0.123629	-5.665	1.47e-08 ***
ma1	-1.533687	0.113285	-13.538	< 2e-16 ***
ma2	0.660454	0.124907	5.288	1.24e-07 ***

ARIMA

More compatible with forecast() and prediction() function

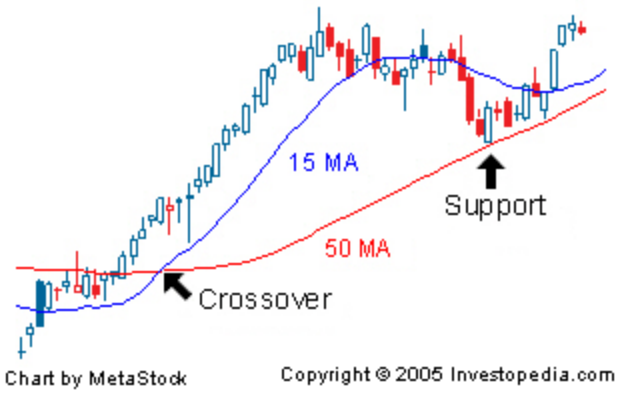
```
>> arima(time-series_data, order = c(0L, 0L, 0L))
      c: order of AR, Series, and MA
ex) order=c(2,0,2) series was 1(1)
      but because of differencing, the "1" part=0 now
```

Moving-average

Motivation

smooth out price action by filtering out the “noise” from random price fluctuations

Day	Closing Price	10-day SMA	Values Used for SMA
1	20		
2	22		
3	24		
4	25		
5	23		
6	26		
7	28		
8	26		
9	29		
10	27	**25**	Average of Day 1 through 10
11	28	**25.8**	Average of Day 2 through 11
12	30	**26.6**	Average of Day 3 through 12
13	27	**26.9**	Average of Day 4 through 13
14	29	**27.3**	Average of Day 5 through 14
15	28	**27.8**	Average of Day 6 through 15



MAs lag current price action because they are ***based on past prices***; the longer the time period for the MA, the greater the lag. Thus a 200-day MA will have a much greater degree of lag than a 20-day MA because it contains prices for the past 200 days.

Residual Analysis

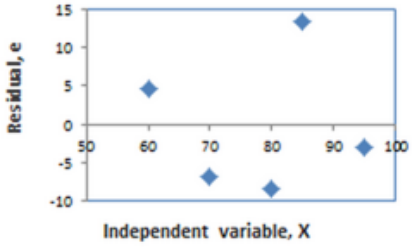
Motivation

To check whether a linear regression model is a good fit.

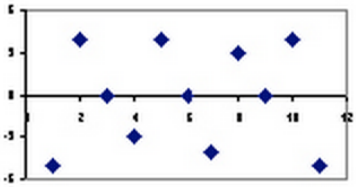
Def

e = y - ŷ (Residual = Observed value - Predicted value)
Both the sum and the mean of the residuals are ALWAYS equal to zero.

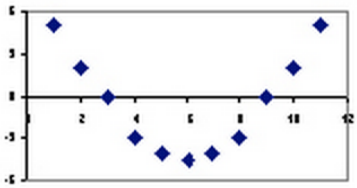
x	60	70	80	85	95
y	70	65	70	95	85
ŷ	65.411	71.849	78.288	81.507	87.945
e	4.589	-6.849	-8.288	13.493	-2.945



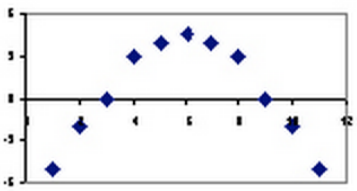
If the points in a residual plot are randomly dispersed around the horizontal axis, a linear regression model is appropriate for the data; otherwise, a non-linear model is more appropriate.



Random pattern



Non-random: U-shaped



Non-random: Inverted U

U-shaped and inverted U suggesting a better fit for a non-linear model

Transformations to Achieve Linearity

Linear transformation:

- multiplying, dividing by a constant
- no change the correlation between x and y

Nonlinear transformation:

- taking the square root of x or the reciprocal of x
- changes linear relationships between variables

Method	Transformation(s)	Regression equation	Predicted value ($\hat{y}$)
Standard linear regression	None	$y = b_0 + b_1x$	$\hat{y} = b_0 + b_1x$
Exponential model	Dependent variable = log(y)	$\log(y) = b_0 + b_1x$	$\hat{y} = 10^{b_0 + b_1x}$
Quadratic model	Dependent variable = sqrt(y)	$\text{sqrt}(y) = b_0 + b_1x$	$\hat{y} = (b_0 + b_1x)^2$
Reciprocal model	Dependent variable = 1/y	$1/y = b_0 + b_1x$	$\hat{y} = 1 / (b_0 + b_1x)$
Logarithmic model	Independent variable = log(x)	$y = b_0 + b_1\log(x)$	$\hat{y} = b_0 + b_1\log(x)$
Power model	Dependent variable = log(y) Independent variable = log(x)	$\log(y) = b_0 + b_1\log(x)$	$\hat{y} = 10^{b_0 + b_1\log(x)}$

How to Perform a Transformation to Achieve Linearity

Transforming a data set to enhance linearity is a multi-step, trial-and-error process.

- Conduct a standard regression analysis on the raw data.
- Construct a residual plot.
    - If the plot pattern is random, do not transform data.
    - If the plot pattern is not random, continue.
- Compute the coefficient of determination (R^2).
- Choose a transformation method (see above table).
- Transform the independent variable, dependent variable, or both.
- Conduct a regression analysis, using the transformed variables.
- Compute the coefficient of determination (R^2), based on the transformed variables.
    - If the transformed R^2 is greater than the raw-score R^2, the transformation was successful.
    - If not, try a different transformation method.

Holt-Winters Filtering

Whereas in the simple moving average the past observations are weighted equally, exponential smoothing assigns exponentially decreasing weights over time.

Prediction vs. Forecast in R

"Forecasting would be a subset of prediction. Any time you predict into the future it is a forecast. All forecasts are predictions, but not all predictions are forecasts, as when you would use regression to explain the relationship between two variables."

```
>> predict(object, n.ahead =..., newxreg = NULL, se.fit = TRUE, ...)
      object: the result of an arima fit.
      n.ahead: the number of steps ahead for which prediction is required
      ex) n.ahead = (length(infy_ret) - (0.9 * length(infy_ret))


>> forecast((object, h =…))
      object: a time series for which forecasts are required
      h: number of periods for forecasting
```

Reference

- Prediction on R
- Prediction on R cnt
- quantmod
- SparkR
- Stationary vs. Non-stationary Data
- Blog
- Differencing
- Autocorrelation equation
- How to interpret autocorrelation
- Time Series Basics
- Residual Analysis
- Transformations to Achieve Linearity
- Correlation Coefficient
- R^2 Coefficient of determination
- R^2 explanation
- Moving Average model