A Primer on Time Series

(One Way To Deal With Time Correlated Data)

Jimmy Horine

August 22, 2015

A Reminder

"Essentially, all models are wrong, but some are useful"

- Box

Beer Data

The beer data as seen on our Meetup.com website found on: https://datamarket.com/data/set/22ry/quarterly-beer-production-in-australia-megalitres-march-1956-june-1994

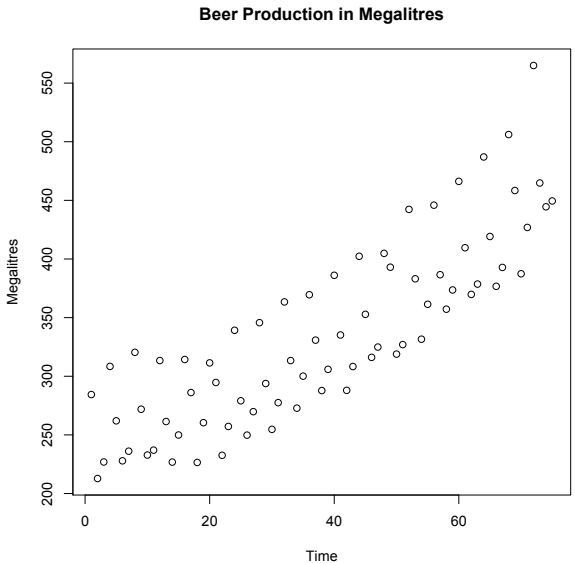


Figure: Mmmm Beer

Usual (Linear Model) Assumptions

In the usual regression setting, we may (and shall) assume:

- The observations are independent (from each other) and identically distributed
- The errors terms are IID *normal* with mean $\mu = 0$ and constant variance $\sigma^2 < \infty$
- Some sort of (intrinsically) linear relationship

With population model:

$$\mathbf{Y} = \boldsymbol{\beta}\mathbf{X} + \boldsymbol{\epsilon}$$

And "best" predictors (under the Gauss-Markov assumptions)

$$\hat{\boldsymbol{\beta}} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{y}$$

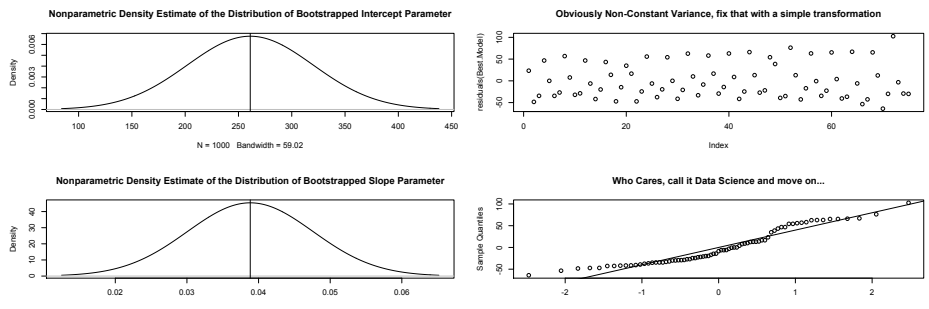
Lets Try A Linear Regression

The fitted population model is:

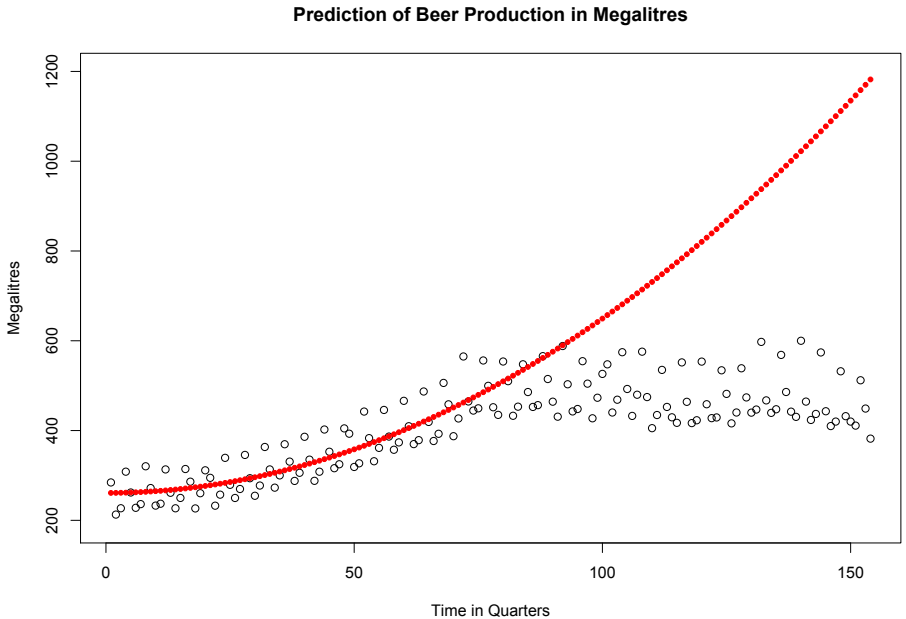
$$\text{Estimated Quarterly Beer Production} = 261.04917 + 0.038849(\mathit{Time})^2$$

Or, using the bootstrap with 1000 samples using 2.146 seconds of computing time:

$$\text{Estimated Quarterly Beer Production}^* = 261.0492 + 0.0388394(\mathit{Time})^2$$



Predictions Using the Best Linear Model



BAD CODE! DO NOT USE

```
#Beer Data Code.....
#This is what you SHOULD NOT DO!
#This is what you SHOULD NOT DO!
#This is what you SHOULD NOT DO!
#This is what you SHOULD NOT DO!

beer <- read.table("/Users/jimmy/Desktop/beer_data.csv", sep=",", header=T)
head(beer)
Beer <- cbind(1:nrow(beer),beer$Megalitres)
Beer2 <- cbind(1:nrow(beer),beer$Megalitres);colnames(Beer2)=c("Time", "Megalitres")
colnames(Beer)=c("Time", "Megalitres")
Beer <- as.data.frame(Beer[1:75,])
head(Beer)
plot(Beer, main="Beer Production in Megalitres")
Best.Model <- lm(Megalitres~I(Time^2), Beer)
Best.Model
summary(Best.Model)
anova(Best.Model)
par(mfrow=c(2,1))
plot(residuals(Best.Model), main="Obviously Non-Constant Variance, fix that with a simple transformation")
qqnorm(residuals(Best.Model),main="Who Cares, call it Data Science and move on...");abline(mean(residuals(Best.Model)),sd(residuals(Best.Model)))
par(mfrow=c(1,1))


#Just to be EXTRA sure we are doing it correctly, lets bootstrap, because why not, thats what you are supposed to do...
#Just to be EXTRA sure we are doing it correctly, lets bootstrap, because why not, thats what you are supposed to do...
#Can you feel the sarcasm?
#Can you feel the sarcasm?
Beta1 <- numeric()
Beta2 <- numeric()
for (i in 1:1000){
    index.to.use <- sample(c(1:75), size=35, replace=T)
    Beer.to.use <- Beer[index.to.use,]
    Beta1[i] <- coef(lm(Megalitres~I(Time^2), Beer))[1]
    Beta2[i] <- coef(lm(Megalitres~I(Time^2), Beer))[2]

}
plot(density(Beta1), main="Nonparametric Density Estimate of the Distribution of Bootstrapped Intercept Parameter");abline(v=median(Beta1))
median(Beta1)
plot(density(Beta2), main="Nonparametric Density Estimate of the Distribution of Bootstrapped Slope Parameter");abline(v=median(Beta2))
median(Beta2)


#So we want to predict a few months out because that is reasonable after all...

New.Data <- data.frame(Time=c(1:154))

My.Predictions <- predict(Best.Model, newdata = New.Data); My.Predictions
plot(Beer2, xlab="Time in Quarters", ylim=c(190, 1200), main="Prediction of Beer Production in Megalitres");points(My.Predictions, col="red", pch=20)
```

Figure: BAD CODE DO NOT USE FOR REAL DATA ANALYSIS. BAD DATA SCIENTISTS, BAD!

The Beer Data

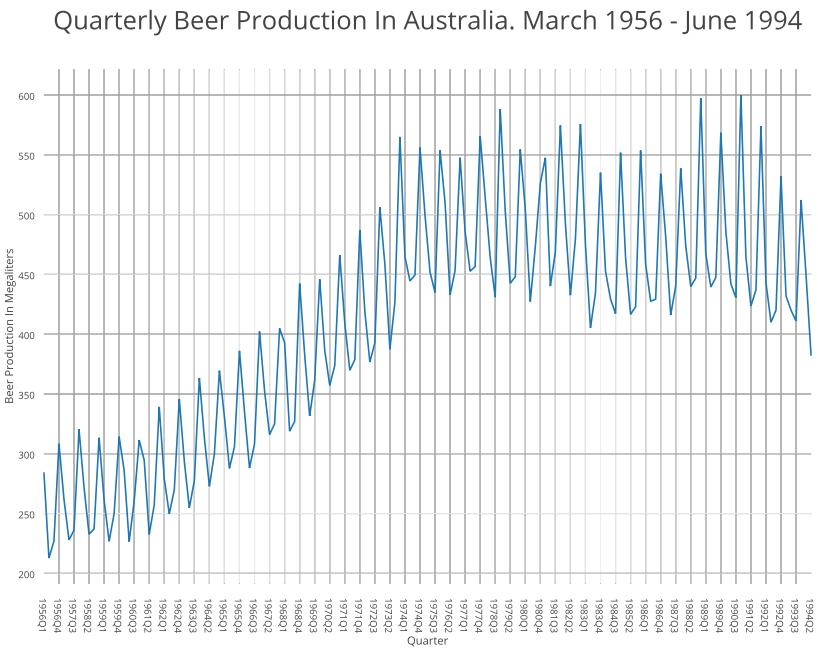


Figure: Mmm Beer...again

Def^{n}: A Time Series

Formally a Time Series is a stochastic (random) process whose *regularly-spaced* realizations (random variables) are indexed by an ordered set T, (usually $\mathbb{N}$).

$\Longrightarrow$ What does that mean?

We will index EACH observation from the random process that we are interested in by the time we observe it, and record the value. So at time T=$\{1, 2, 3, \ldots\}$, we record a measurement.

Some Time Series Preliminaries

Time Series, like many other modeling paradigms has a few basic assumptions.

- First, we ASSUME the observations to be correlated with each-other.
- Second, we assume (hope) the process is *stationary*. That is, the process has zero mean (or if not a zero mean, a mean level that may and shall be easily removed from the process), and the second moment of the process depends only on the *lag* between observations.

You may see the notation as: Let X_t be a stationary process with $E(X_t) = 0$, $E(X_t X_t) = \sigma_x^2$, and $E(X_t X_{t-k}) = \gamma_k$.

Notice that the first two moments are free of time! They are assumed to be constant through out the process.

The Autoregressive Model

Autoregressive models are one of the most popular modeling paradigms in time series today. The autoregressive process of order P is defined as:

$$X_t = \alpha_1 X_{t-1} + \alpha_2 X_{t-2} + \ldots + \alpha_p X_{t-p} + Z_t, \qquad \text{where } Z_t \sim N(0, \sigma_Z^2)$$

Remember from the previous slide, when modeling an autoregressive process, we are concerned with weak (second order) stationary. That is, where we have a zero-mean process, or one that can be arithmetically centered to have zero mean, AND where the autocovariances do not evolve over time.

As a mathematical consequence, the AR(P) process is considered to be (casual) stationary if the roots of the back-shift operator polynomial are outside of the unit circle on the complex plane. That is the function

$$\phi(B) = 1 - \sum_{k=1}^{p} \alpha_k B^k$$

where the α's are the autoregressive weights, has roots with modulus greater than 1.

The Moving Average Model

Alternatively, we may model the process as a linear combination of past noise. Formally, a moving average process of order Q is defined as

$$X_t = Z_t + \beta_1 Z_{t-1} + \beta_2 Z_{t-2} + \ldots + \beta_p Z_{t-q}, \qquad \text{where } Z_t \sim WN(0, \sigma_Z^2)$$

Estimation of the model parameters for a MA(Q) process is messy as the shocks or innovations or errors are NOT observable. Moreover, It may be shown that an any MA(Q) may be represented as an AR(P) (and vice versa). Therefore, we may (and shall) approximate any MA(Q) process with an AR(P) of sufficiently high order.

Besides, the theory dealing with an AR(P) is just more fun.

Model Identifiably

We intentionally glossed over some key points earlier. Lets revisit them now. Most of the Greek letters used here are just definitions and notation - nothing too wild.
Let X_t be, at least, a weakly stationary process with

- Mean $E(X_t) = 0$
- Variance $E(X_t X_t) = \sigma_x^2$
- Autocovariance $E(X_t X_{t-k}) = \gamma_k$.

We will normalize the autocovariance by dividing out by the autocovariance at lag 0 to obtain the *Autocorrelation Function*.

Additionally we call the *partial* autocorrelation function as

$$pACF_k = \frac{Corr(X_t, X_{t+k} | X_{t+1}, \ldots, X_{t+k-1})}{Cov(X_t, X_t)}$$

It may be easily shown that:

- The pACF "Shuts Off" at lag P+1 for a true AR(P) process.
- The ACF "Shuts Off" at lag Q+1 for a true MA(Q) process.

Estimation

Estimation of model parameters is easily carried out through the use of the Yule-Walker Equations, Ordinary Least Squares, or (and more complicatedly) Maximum Likelihood Estimation. Most software packages, including R, allow the user to specify which estimation procedure to use.

The Yule-Walker Equations are formulated as multiplying random variables of the process by lagged random variables, taking expectation, and dividing through by N-1. You may choose to divide the system again by the constant $E(X_t X_t) =< X_t X_t >= \gamma_0$ and instead work with correlations, opposed to covariances.

$$\begin{bmatrix} \gamma_1 \\ \gamma_2 \\ \gamma_3 \\ \vdots \\ \gamma_p \end{bmatrix} = \begin{bmatrix} \gamma_0 & \gamma_{-1} & \gamma_{-2} & \cdots \\ \gamma_1 & \gamma_0 & \gamma_{-1} & \cdots \\ \gamma_2 & \gamma_1 & \gamma_0 & \cdots \\ \vdots & \vdots & \vdots & \ddots \\ \gamma_{p-1} & \gamma_{p-2} & \gamma_{p-3} & \cdots \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \vdots \\ \alpha_p \end{bmatrix}$$

$$\boldsymbol{\gamma} = \boldsymbol{\Gamma}\boldsymbol{\alpha}$$

$$\boldsymbol{\alpha} = \boldsymbol{\Gamma}^{-1}\boldsymbol{\gamma}$$

Note: $\boldsymbol{\Gamma}$ is Full Rank, Symmetric, and Toeplitz.

Identification

The "Shut Off" behavior of the ACF and pACF for generating processes gives us Autoregressive and Moving Average "Signature."

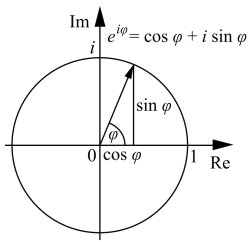
- AR(P):
  - Long damped sinusoidal or exponential decay in the ACF.
  - Abrupt Shut Off in the pACF at order P.
- MA(Q):
  - Long damped sinusoidal or exponential decay in the pACF.
  - Abrupt Shut Off in the ACF and order Q.
- ARMA(P,Q): Long damped sinusoidal or exponential decay *potentially* seen in both ACF and pACF.

Identification and Implications of the Backshift Operator Polynomial

Recall the AR backshift polynomial:

$$\Phi(B) = 1 - \sum_{k=1}^{p} \alpha_k B^k$$

where the α's are the autoregressive weights. This polynomial has roots in $\mathbb{C}$.



The *modulus* of a root relates to the strength of that component of process, the closer to 1 in modulus the root is, the closer to non-invertible the system becomes and thus more explosive. The angle ψ yields a periodicity. Remember that roots in the complex plane arrive in conjugate pairs, except for when $\psi = 0$ or π, in which case we see no periodicity or infinite periodicity.

An AR(P) process will have P-many roots-pairs and periodicities!

Identification and The Sample Spectrum

Moving from the Time Domain to the Frequency Domain is facilitated through the use of the Fourier Transform of the *data*. Taking the modulus squared Fourier transform of the data IS the same as taking the Fourier transform of the autocovarinace sequence.

Let the Sample Spectral Density or Periodogram be given as:

$$I_{n\omega_k} = \frac{1}{n}\left|\sum_{n=1}^{N} X_n e^{-i\omega_k t}\right|^2$$

$$= \sum_{|h|<N} \gamma_h e^{-ih\omega_k}$$

Where $\omega_k = \frac{2\pi}{n}$

The theoretical spectral density is given by:

$$f(\omega) = \frac{\sigma_z^2}{2\pi}\frac{|\Theta e^{-i\omega}|^2}{|\Phi e^{-i\omega}|^2}$$

Where Θ is the MA backshift operator polynomial and Φ is the AR backshift operator polynomial. Roots in Θ yield dips in the spectral density, and roots in Φ yield peaks!

The Beer Data One Last Time

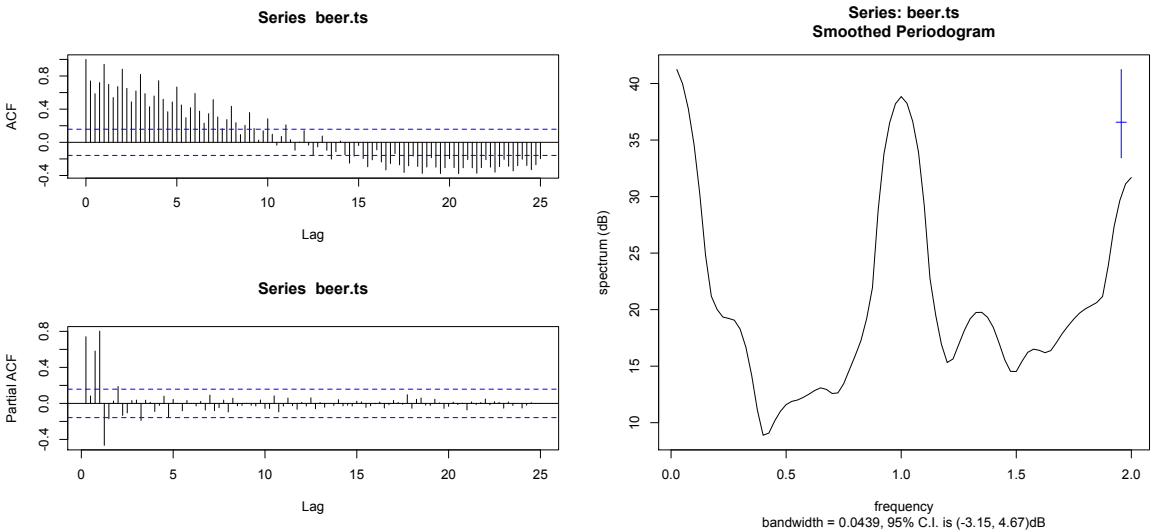


Figure: Sample ACF/pACF and Sample Periodogram

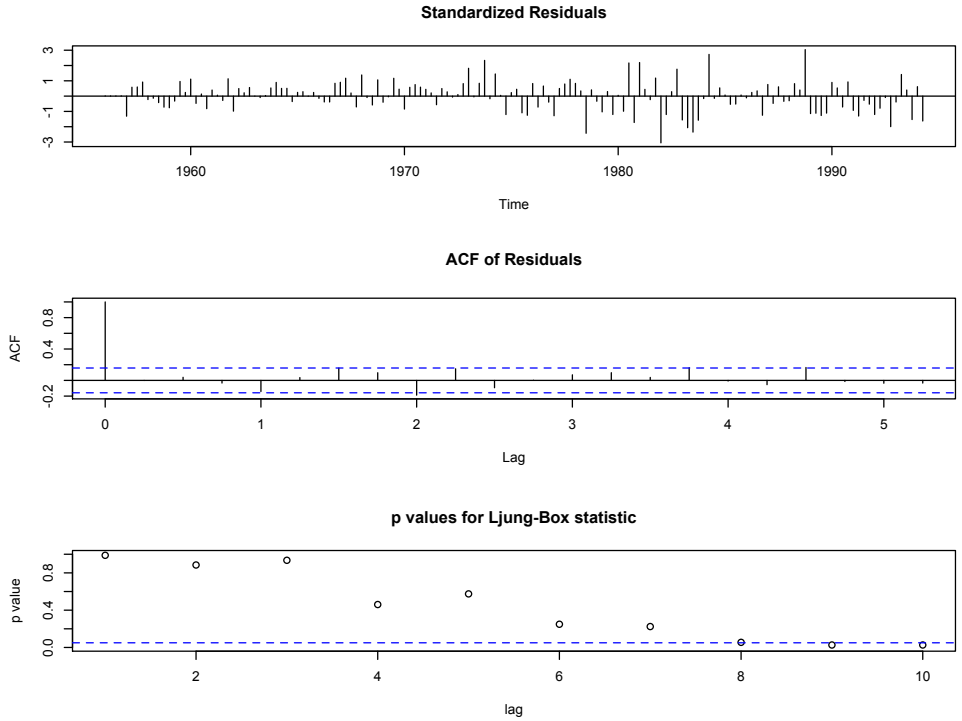
The Fitted Beer Model

```
Series: beer.ts
ARIMA(4,0,1)(0,1,0)[4] with drift

Coefficients:
         ar1     ar2     ar3      ar4      ma1   drift
      0.0680  0.3024  0.3829  -0.1814  -0.0911  1.1036
s.e.  0.3008  0.0799  0.1069   0.1434   0.2973  0.7556

sigma^2 estimated as 312.5:  log likelihood=-624.95
AIC=1263.9   AICc=1264.69   BIC=1284.97
```

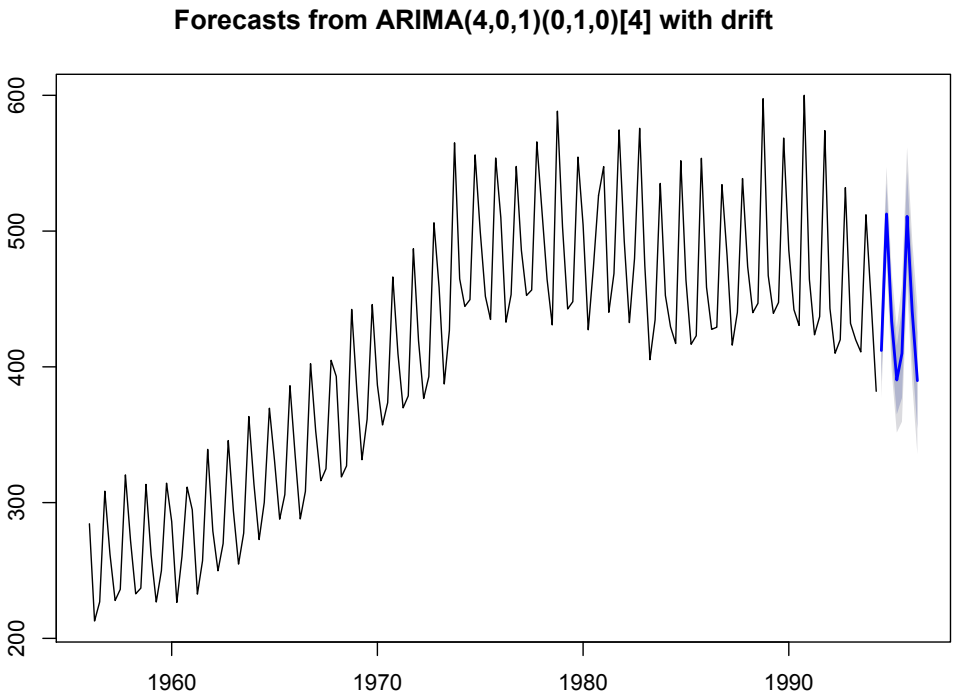
Beer R Code

```
#Time Series Beer Code!
#install.packages("forecast")
require(forecast)
beer <- read.table("/Users/jimmy/Desktop/beer_data.csv", sep=",", header=T)
head(beer)

beer.ts <- ts(beer$Megalitres, frequency=4, start=c(1956,1))
plot.ts(beer.ts)

#Some Sample Statistics
par(mfrow=c(2,1))
acf(beer.ts, lag.max=100)
pacf(beer.ts,lag.max=100)
par(mfrow=c(1,1))
spec.pgram(beer.ts, spans=c(5,5), taper=.15, log="dB")

#Examining the data for trend components
beer.decomp <- decompose(beer.ts, type="additive")
plot(beer.decomp)

#fitting an ARIMA model using the forecast package
beer.model <- Arima(beer.ts, order=c(4,0,1), seasonal=c(0,1,0), include.drift=TRUE)
beer.model
tsdiag(beer.model)
beer.forecasts <- forecast(beer.model)
plot(beer.forecasts)
```

Beer Model Diagnostics



Beer Model Forecasts



Tons 'o Data

Huge ammount of time series data to play with:

`https://datamarket.com/data/list/?q=provider:tsdl`