Forecasting

Jeffrey Leek, Assistant Professor of Biostatistics

May 18, 2016

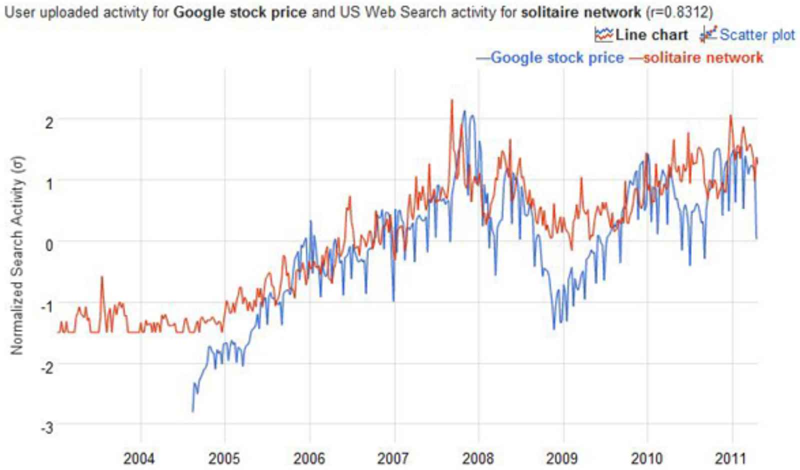
Time series data

https://www.google.com/finance

What is different?

- Data are dependent over time
- Specific pattern types
- Trends - long term increase or decrease
- Seasonal patterns - patterns related to time of week, month, year, etc.
- Cycles - patterns that rise and fall periodically
- Subsampling into training/test is more complicated
- Similar issues arise in spatial data
- Dependency between nearby observations
- Location specific effects
- Typically goal is to predict one or more observations into the future.
- All standard predictions can be used (with caution!)

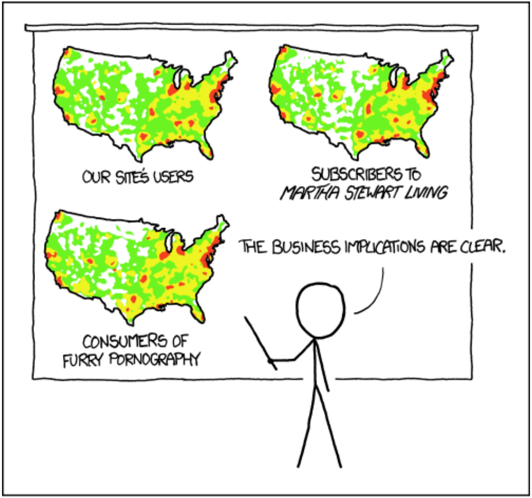
Beware spurious correlations!



http://www.google.com/trends/correlate

http://www.newscientist.com/blogs/onepercent/2011/05/google-correlate-passes-our-we.html

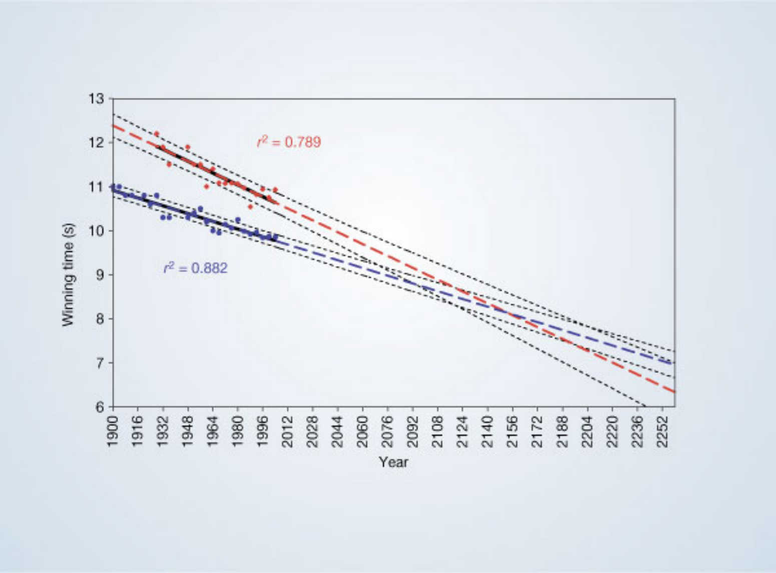
Also common in geographic analyses



PET PEEVE #208:
GEOGRAPHIC PROFILE MAPS WHICH ARE
BASICALLY JUST POPULATION MAPS

http://xkcd.com/1138/

Beware extrapolation!



http://www.nature.com/nature/journal/v431/n7008/full/431525a.html

Google data

```
library(quantmod)
from.dat <- as.Date("01/01/08", format="%m/%d/%y")
to.dat <- as.Date("12/31/13", format="%m/%d/%y")
getSymbols("GOOG", src="google", from = from.dat, to = to.c
head(GOOG)
```

Summarize monthly and store as time series

```
mGoog <- to.monthly(GOOG)
googOpen <- Op(mGoog)
ts1 <- ts(googOpen,frequency=12)
plot(ts1,xlab="Years+1", ylab="GOOG")
```

Example time series decomposition

- **Trend** - Consistently increasing pattern over time
- **Seasonal** - When there is a pattern over a fixed period of time that recurs.
- **Cyclic** - When data rises and falls over non fixed periods

`https://www.otexts.org/fpp/6/1`

Decompose a time series into parts

```
plot(decompose(ts1),xlab="Years+1")
```

Training and test sets

```r
ts1Train <- window(ts1,start=1,end=5)
ts1Test <- window(ts1,start=5,end=(7-0.01))
ts1Train
```

Simple moving average

$$Y_t = \frac{1}{2 * k + 1} \sum_{j=-k}^{k} y_{t+j}$$

```
plot(ts1Train)
lines(ma(ts1Train,order=3),col="red")
```

Exponential smoothing

Example - simple exponential smoothing

$$\hat{y}_{t+1} = \alpha y_t + (1-\alpha)\hat{y}_{t-1}$$

		Seasonal Component	
Trend Component	N (None)	A (Additive)	M (Multiplicative)
N (None)	(N,N)	(N,A)	(N,M)
A (Additive)	(A,N)	(A,A)	(A,M)
A_d (Additive damped)	(A_d,N)	(A_d,A)	(A_d,M)
M (Multiplicative)	(M,N)	(M,A)	(M,M)
M_d (Multiplicative damped)	(M_d,N)	(M_d,A)	(M_d,M)

`https://www.otexts.org/fpp/7/6`

Exponential smoothing

```r
ets1 <- ets(ts1Train,model="MMM")
fcast <- forecast(ets1)
plot(fcast); lines(ts1Test,col="red")
```

Get the accuracy

```
accuracy(fcast,ts1Test)
```

Notes and further resources

- Forecasting and timeseries prediction is an entire field
- Rob Hyndman's Forecasting: principles and practice is a good place to start
- Cautions
- Be wary of spurious correlations
- Be careful how far you predict (extrapolation)
- Be wary of dependencies over time
- See quantmod or quandl packages for finance-related problems.