# STATS 207: Time Series Analysis
# Autumn 2020

## Lecture 1: Course outline, Examples of Time Series Data, Models for Time Series Data.

Dr. Alon Kipnis

Slides credit: David Donoho, Dominik Rothenhäusler

September 14th 2020

# Outline of first lecture

1. Course outline and organizational matters.
2. Examples of time series data.
3. Tentative list of topics.
4. Models for time series Data.

# Course outline and organizational matters

# Organizational matters

- **Instructor**: Alon Kipnis
- **Lectures**: 10:00 - 11:20 am Mon, Wed, using **Zoom**.
- **Teaching Assistant**: Zijun Gao and Anav Sood.
- **Course Staff Email Address**: stats207-aut2021-all@lists.stanford.edu
- **Online Office Hours (aka Office Chats, aka Coffee Breaks)**: 11:20 - 12:20 Mon, Wed, using **Zoom**. https://stanford.zoom.us/my/kipnisal
- **TA Online Office Hours**: Details will be posted on Canvas.

## Course pages

1. Lecture material on **Canvas** (slides, sample R code, homework etc.)
2. Other course-related announcements on **Canvas** (https://canvas.stanford.edu/)
3. Discussions on **Piazza** (https://piazza.com/stanford/fall2020/stats207/home)
4. Home assignments and grades will be posted on **Gradescope** (https://www.gradescope.com/courses/173400)

# COVID-19 and online learning

- Online learning is new to this class.
- The quarter is shorter than usual (10 compared to 12 weeks).
- Let us know if you have suggestions on how to improve your learning experience.
- **We are here to help**. We look forward to seeing you in our virtual office hours.

# Recording

- Lectures will be recorded. They will be available on **Canvas**.
- I strongly encourage you to attend the class live.

# Time Zone

- **West-coast** time (aka PT, usually UTC-08:00)
- If you are currently not in the US, please let us know what time zone you're in. You can reach us at stats207-aut2021-all@lists.stanford.edu .
- Depending on this feedback, we may change some of our office hours to address accessibility issues due to time zone differences.

## Prerequisites:

- **Elementary statistics** at level of STATS 200 (correlation, maximum likelihood, least squares, confidence intervals,. . . )
- **Elementary probability** at level of STAT 116 (random variables, independence, correlation, joint distributions, . . . )
- Some background on **complex numbers** (not mandatory)
- Basic programming skills in **R** (not mandatory)

## Textbook and R

- The main textbook:
  **Shumway & Stoffer, "Time Series Analysis and its Applications"** (henceforth [Shumway & Stoffer] ).
- Available at http://www.stat.pitt.edu/stoffer/tsa4
  (visit website **now**!)
- All figures in the book are reproducible at the book website.

- The programming language of the course is **R**.
  Available at http://cran.r-project.org
- You may use a different programming language **at your own risk!**
- Why you should use **R** for data science:
  - 'ggplot'
  - 'tidyverse'

# Homeworks

- Constitute 80% of the final grade.
- Mix of theoretical (pen and paper) and computer exercises.
- Will be posted posted every **two weeks**.
- All homework needs to be submitted via **Gradescope**.
- Homework collaboration policy:
  - Every student must first attempt **all** problems **individually**.
  - You may discuss a homework assignment with up to **two classmates**.
  - Each student must write up his/her **own** solutions individually and explicitly **name any collaborators** at the top of the homework.
- **Regrade requests** must be submitted within **one week** after grading has been published.
- Regrading requests are submitted via Gradescope.

## Assessment and grading:

- **Grading:** 80% regular homework assignments, 20% take home exam.
- Take-home exam:
  - About **2 hours** time-limit.
  - Can access at your free time during the last week of classes 11/16-11/20.
  - Ideology: easy to get near perfect grade if you review class material and home assignments **before** starting the exam.

# Online Learning Community

- We encourage discussions between classmates, either on Piazza or elsewhere.
- We encourage you to attend our virtual office hours.
- Please send us interesting related dataset and articles so we can share with everyone (‘Medium’ and ’Toward Data Science’ are nice sources).

# Examples of Time Series Data

## Examples

- Johnson and Johnson quarterly earning
- Global Temperature Deviations
- Speech Data
- Dow-Jones Industrial Average
- Fish Population and El-Ninõ
- fMRI Data
- Daily New Cases of Covid-19
- Air Quality Data

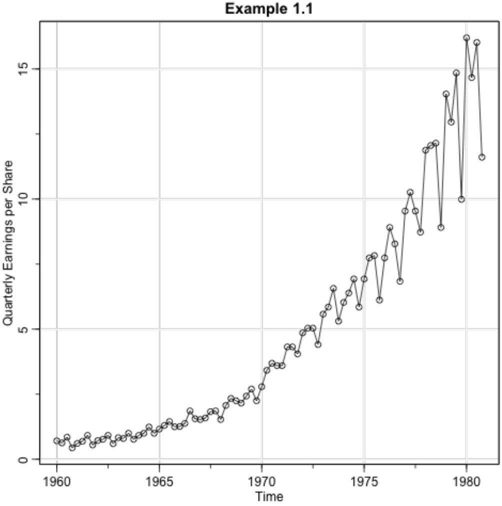
## Example 1.1

Example 1.1 in [Shumway & Stoffer] : Johnson and Johnson earnings

- $N = 84$ data points.
- Earning per share of JnJ stock.
- Quarterly numbers 21 years of data.
- The data:

| year | Qtr1 | Qtr2 | Qtr3 | Qtr4 |
|---|---|---|---|---|
| 1960 | 0.71 | 0.63 | 0.85 | 0.44 |
| 1961 | 0.61 | 0.69 | 0.92 | 0.55 |
| 1962 | 0.72 | 0.77 | 0.92 | 0.60 |
| 1963 | 0.83 | 0.80 | 1.00 | 0.77 |
| 1964 | 0.92 | 1.00 | 1.24 | 1.00 |
| $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ |
| 1980 | 16.20 | 14.67 | 16.02 | 11.61 |

# Example 1.1: The Plot

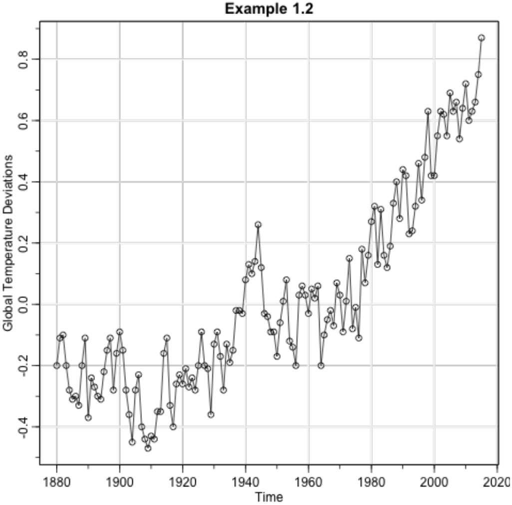


# Example 1.1: The Code

```
plot(jj, type="o", ylab="Quarterly Earnings per Share",
        main="Example 1.1")
```

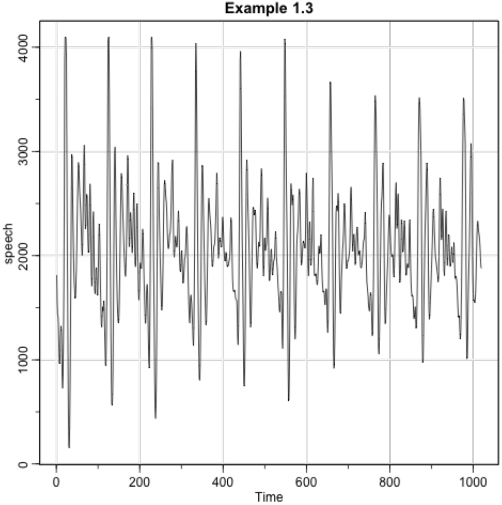
Code of all the examples from [Shumway & Stoffer] are available at https://www.stat.pitt.edu/stoffer/tsa4/Rexamples.htm

# Example 1.2: Global Temperature Deviations

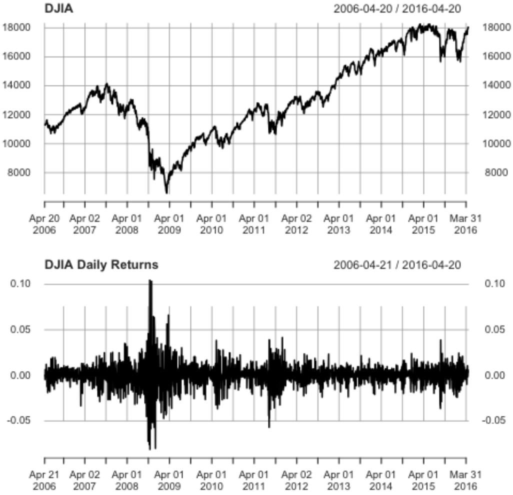


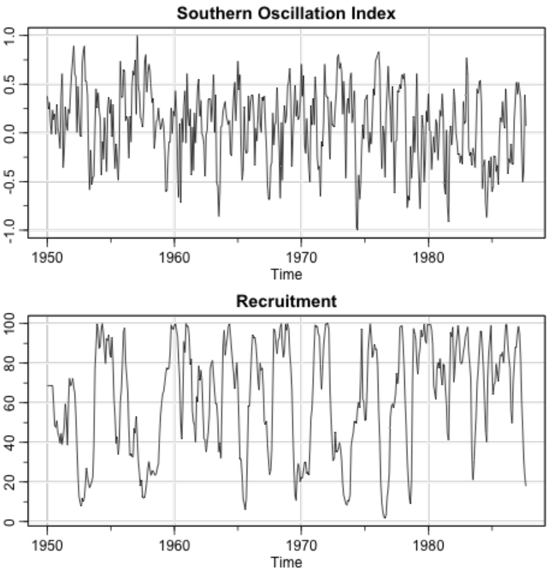
# Example 1.3: Speech Data

Speech recording of the syllable aaa...hhh sampled at $10,000$ points per second with $n = 1020$ points:

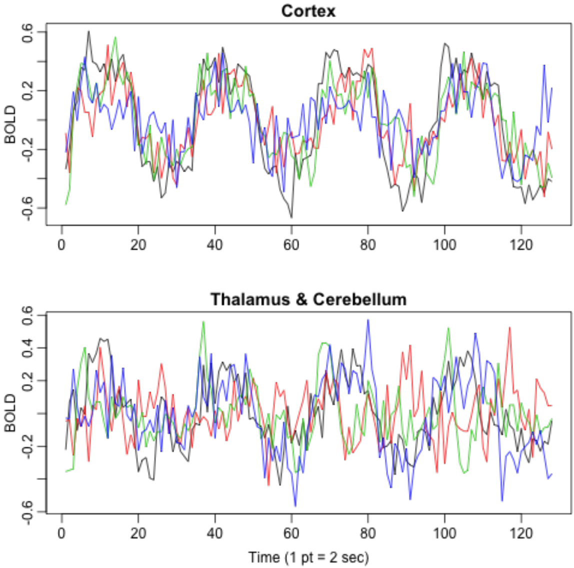


# Example 1.4: Dow-Jones Daily Returns

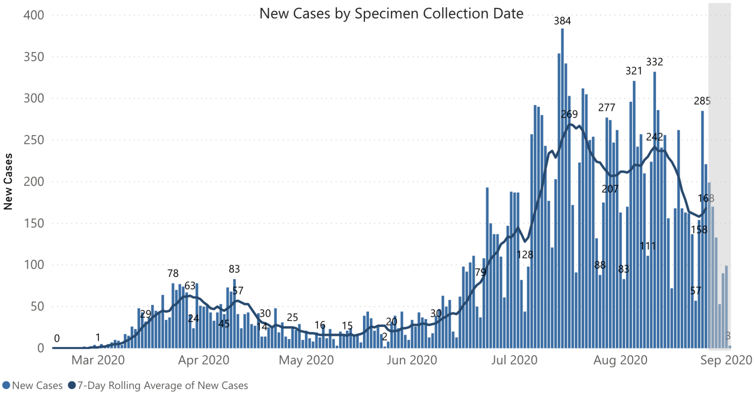



Southern Oscillation Index
1.0
0.5
0.0
-0.5
-1.0
1950
1960
1970
1980
Time
Recruitment
100
80
60
40
20
0
1950
1960
1970
1980
Time

# Example 1.6: fMRI Data

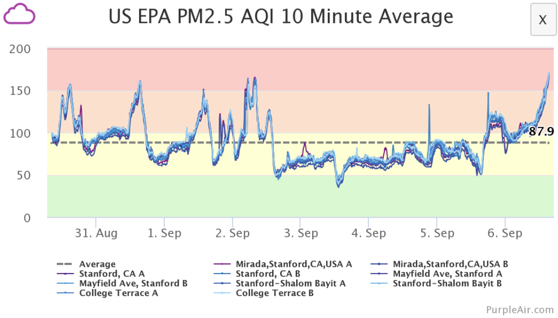


# Example: Daily New Cases of Covid-19 in Santa Clara County




Data: Santa Clara County Covid-19 Cases Dashboard https://www.sccgov.org/sites/covid19/Pages/dashboard-cases.aspx

# Example: Air Quality




Source: PurpleAir, LLC
https://www.purpleair.com/map?opt=1/mAQI/a10/cC0#12.43/37.42184/-122.17378

## Other Examples

- Average Happiness for Twitter
  http://hedonometer.org/timeseries/en_all/
- Google Trends https://trends.google.com/trends/explore?date=all&geo=US&q=Time%20Series,%2Fm%2F041m_j

# Attributes of Time Series

- Scalar, bivaraite, vectorial
- Regular, irregular
- Sampling frequency: yearly/quarterly/monthly/daily/.../millisecond/../microsecond/...
- Structures:
  - Trend
  - Seasonality
  - Periodicity
- Autocorrelation and Cross-correlation (TBD)

# STATS 200 (Theory of Statistics) vs. STAT 207 (Time Series)

**Simple random sampling:**
$n$ independent, identically dist. observations.

$\Rightarrow$ Learn population distribution as $n \to \infty$.

**Time series:**
$n$ not independent and/or identically dist. observations.

$\Rightarrow$ Explore serial structure to learn dependence as $n \to \infty$.

**Primary objectives in time series analysis:**

- Develop **mathematical models** that provide plausible descriptions for sample time series data.
- Develop **estimation and prediction** for these models.

# Tentative list of topics

## Tentative list of topics

1. **Models for time series data**: mean, autocorrelation, cross-correlation functions, stationarity, estimation of correlation
2. **Trend and seasonality**: trend and seasonality models, heteroscedasticity, variance stabilization
3. **Time series regression**: classical regression in the TS context, model complexity
4. **Prediction and estimation** estimating model parameters, prediction, partial autocorrelation function
5. **Non-linear models**: ARCH, GARCH, stochastic volatility (possibly a guest lecture)
6. **Spectral Analysis**: periodogram, spectral density, linear filtering, cross-spectra
7. **High-dimensional time-series models**: VAR, VARMA, **Prophet** (probably a guest lecture)
8. **State-space models**: Linear state-space models, prediction, Kalman Filter

The full syllabus is on **Canvas**

# Models for Time Series Data

# Stochastic Processes

Definition: A (discrete-time) *stochastic process* is a set of random variables indexed by $\mathbb{N} = \{1, 2, \ldots\}$. Equivalent symbols:

$$(x_t), \quad \{x_t\}_{t=1,2,\ldots}, \quad \{x_t\}_{t\in\mathbb{N}}$$

Definition: The *realization* of a stochastic process are the observed values (sample).

We use the term *time series* to indicate one of three object (the interpretation depends on the context):

1. A generic stochastic process
2. A particular realization of the stochastic process
3. A data set with one measurement per unit time

## White Noise and Moving Average

- **White noise process** $(w_t)$: $w_t \overset{iid}{\sim} P$ for some distribution $P$ with mean 0 and variance $\sigma^2$.

  Important special case: $P = \mathcal{N}(0, \sigma^2)$ (white Gaussian noise).

- **Moving average**. For example

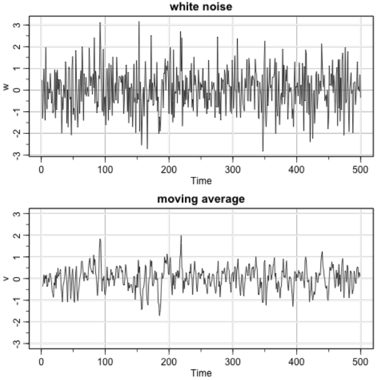
$$v_t = \frac{1}{3}(w_{t-1} + w_t + w_{t+1}),$$

  where $w_t$ is Gaussian noise.

# White Noise and Moving Average

The code:

```
w = rnorm(500,0,1) # 500 N(0,1) variates
v = filter(w, sides=2, rep(1/3,3)) # moving average
par(mfrow=c(2,1)) # stack two figures in a row
plot.ts(w, main="white noise")
plot.ts(v, ylim=c(-3,3), main="moving average")
```



# Auto-Regression

- **Auto-regressive processes**. For example

$$x_t = 0.9x_{t-1} + w_t,$$

plus initial conditions.

- **Random Walk** (special case of an auto-regressive process)

$$x_t = x_{t-1} + w_t$$

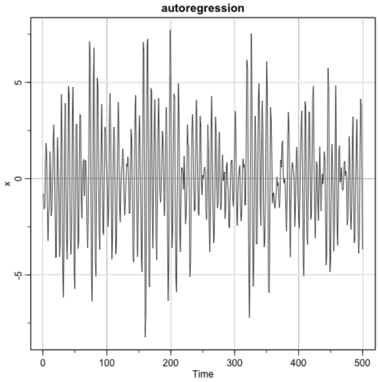
or, with drift,

$$x_t = x_{t-1} + 0.2 + w_t.$$

## The Code

Autoregression:

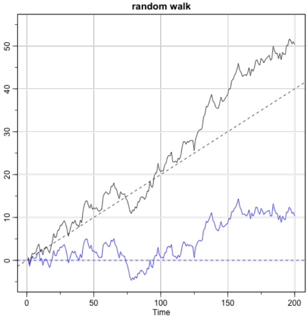
```
w = rnorm(550,0,1) # 50 extra to avoid startup problems
x = filter(w, filter=c(1,-.9), method="recursive")[-(1:50)]
plot.ts(x, main="autoregression")
```



# The Code

Random Walk:

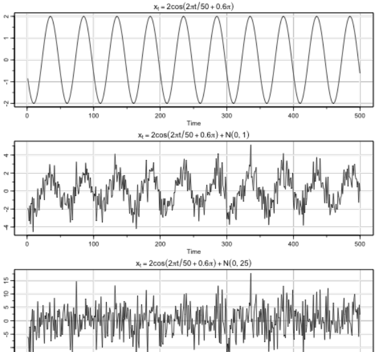
```
set.seed(154) # so you can reproduce the results
w = rnorm(200); x = cumsum(w) # two commands in one line
wd = w +.2;  xd = cumsum(wd)
tsplot(xd, ylim=c(-5,55), main="random walk", ylab="")
lines(x, col=4)
abline(h=0, col=4, lty=2)
abline(a=0, b=.2, lty=2)
```



# Sinusoid in Noise

$$x_t = 2\cos(2\pi t/50 + 0.6\pi) + w_t$$

```
cs = 2*cos(2*pi*(1:500)/50 + .6*pi)
w = rnorm(500,0,1)
par(mfrow=c(3,1), mar=c(3,2,2,1), cex.main=1.5)   # help(par) for info
tsplot(cs, ylab="", main = expression(x[t]==2*cos(2*pi*t/50+.6*pi)))
tsplot(cs + w, ylab="", main =
expression(x[t]==2*cos(2*pi*t/50+.6*pi)+N(0,1)))
tsplot(cs + 5*w, ylab="", main =
expression(x[t]==2*cos(2*pi*t/50+.6*pi)+N(0,25)))
```



# Models for Time Series Data

| Name | Example |
|---|---|
| White noise | $w_t \sim \mathcal{N}(0, \sigma^2)$ |
| Moving Average | $x_t = (w_{t-1} + w_t + w_{t+1})/3$ |
| Autoregression | $x_t = x_{t-1} - 0.9x_{t-2} + w_t$ |
| Random Walk | $x_t = x_{t-1} + w_t$ |
| Sinusoid in noise | $x_t = 2\cos(2\pi t/50 + 0.6\pi) + w_t$ |