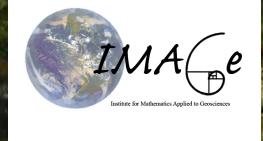


# APPM 2720 Spring 2016

## Lecture 1

Douglas Nychka  
National Center for Atmospheric Research



# Goal

This course is to expose students to data analysis and discovery using techniques from data science.

*Data Science:* is an interdisciplinary field about processes and systems to extract knowledge or insights from data in various forms, either structured or unstructured.

# Statistics is just part of this

A classic intro stats course has things such as

- probability discrete then maybe continuous distribution
- concept of sample verses population
- basic statistics, eg. mean, standard deviation and histograms
- testing for the population mean and confidence intervals
- straight line fitting

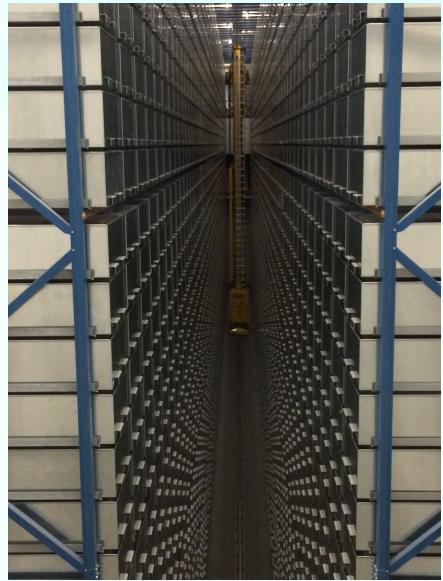
*All these elementary topics are designed to reinforce basic principles in statistics.*

*But they rarely show the value for large and complex data problems!*

# What this course is about

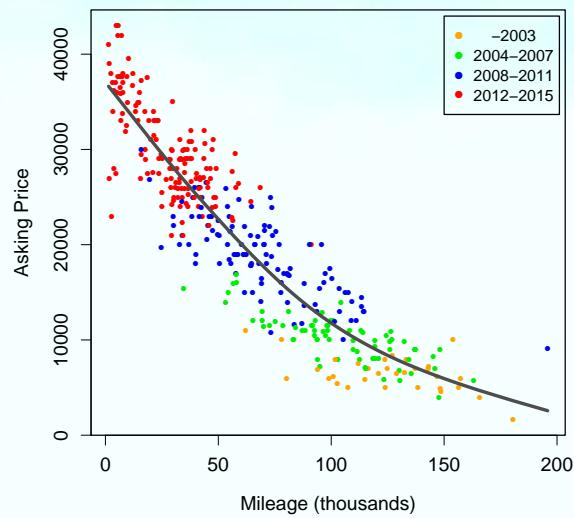
- Confront a dataset based on answering a question.
- Analysis tools and strategies used to reach an answer will be examples of statistical concepts.
- Develop programming skill in R and related programs to look at data.
- Although mathematical formulas will not be used much there still must be an strong element of logical thinking

# Some data examples



# Used Audi A4 prices

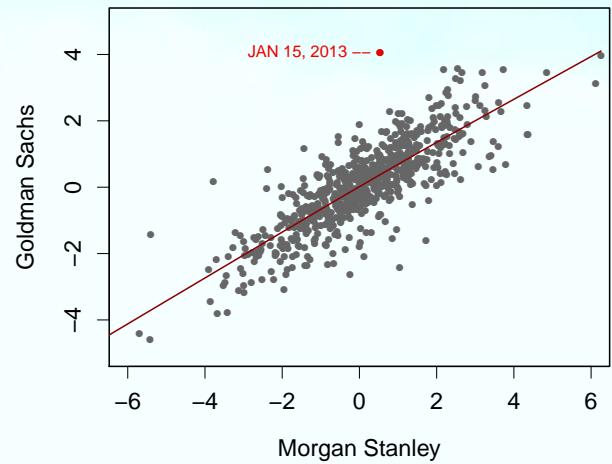
Used Audi A4 prices cars.com



How do you quantify the tradeoff between older cars and cheaper prices?

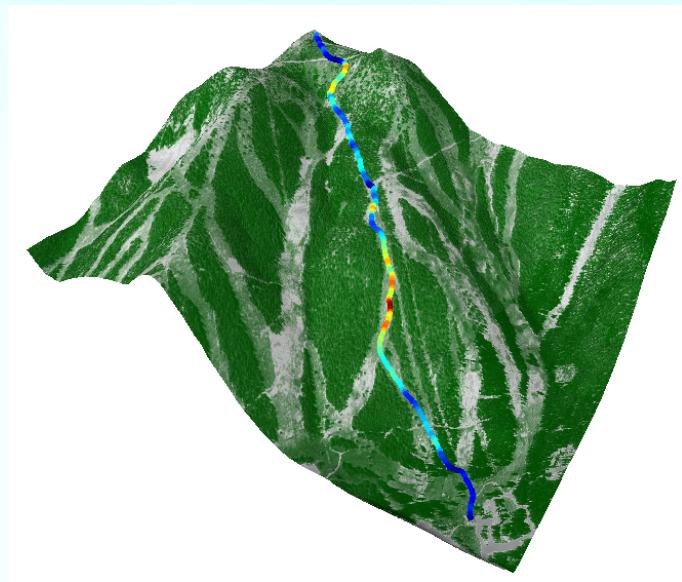
# Stock prices

Daily percent return Oct 2012 – Sep 2015



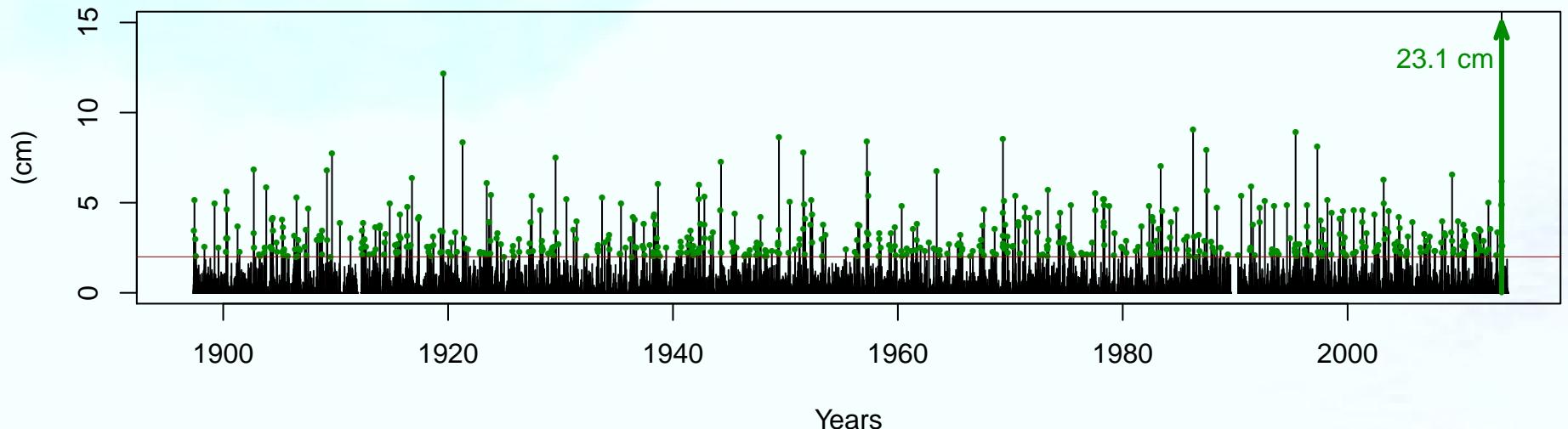
How well do Goldman Sachs and Morgan Stanley stocks track each other?

# Mary Jane Resort



Where is the steepest part of a ski run?

# Boulder daily rainfall



What is the probability of rainfall in Boulder exceeding 8 cm ( about 3.2 inches) in a day?

# Digital images: 100 Weddings



The average image



*wedding*

J. Salavon  
*Cabinet 15*

See Jason Salavon on Wikipedia

# An example of R code:

```
1:10  
  
## [1] 1 2 3 4 5 6 7 8 9 10  
  
mean(1:10)  
  
## [1] 5.5
```

# Computation

```
n<-1:50
```

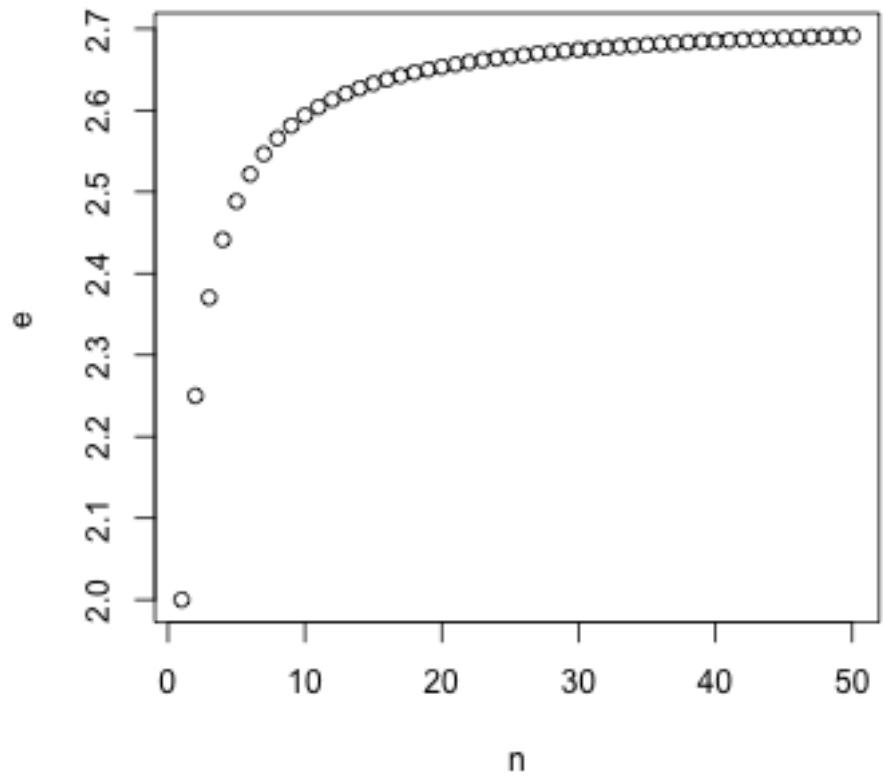
```
e<-(1+ 1/n)^n
```

```
print(e)
```

```
## [1] 2.000000 2.250000 2.370370 2.441406 2.488320 2.521626 2.546500  
## [8] 2.565785 2.581175 2.593742 2.604199 2.613035 2.620601 2.627152  
## [15] 2.632879 2.637928 2.642414 2.646426 2.650034 2.653298 2.656263  
## [22] 2.658970 2.661450 2.663731 2.665836 2.667785 2.669594 2.671278  
## [29] 2.672849 2.674319 2.675696 2.676990 2.678208 2.679355 2.680439  
## [36] 2.681464 2.682435 2.683357 2.684232 2.685064 2.685856 2.686612  
## [43] 2.687333 2.688022 2.688681 2.689312 2.689917 2.690497 2.691053  
## [50] 2.691588
```

# Graphics

```
plot( n,e)
```



# Thank you!

Questions?

