Technical Session

DATA SCIENCE IN ASSET MANAGEMENT

Tuesday, October 19

DATA SCIENCE IN ASSET MANAGEMENT

WELCOME TO PART 1: TECHNICAL SESSION (55 MINUTES)

Don Hejna, CFA, CAIA, FDP

HISTORY

This Webinar arose from what was to be an in-person seminar in Minnesota last summer.

It grew as various CFA Chapter's sponsored the event topping over 400 registered attendees.

Thank you to our sponsors, speakers, and guests attending from across the US & Canada.

A quick word about the distinctions and focus of the various charters:

- The CFA charter gives you expertise and real-world skills in investment analysis and is one of the highest distinctions in the investment management profession.
- The CAIA charter is the globally-recognized credential for professionals managing, analyzing, distributing, or regulating alternative investments.
- The FDP charter is the globally recognized credential for professionals managing, analyzing, translating, and distributing data in finance.

WELCOME TO PART 1: TECHNICAL SESSION

Three speakers today:

Chapter 1: Creating custom tools in Python for manager style analysis.

Chapter 2: Driving commercial tools and data for investment performance gains.

Chapter 3: Leveraging the power of R and open-source tools for financial analysis.
& Gains from alternative data.

PART 2 will begin

10:30 AM PACIFIC

12:30 PM CENTRAL

1:30 PM EASTERN

USING R FOR FINANCE AND PORTFOLIO CONSTRUCTION

LEVERAGING THE POWER OF OPEN-SOURCE SOFTWARE

Don Hejna, CFA, CAIA, FDP

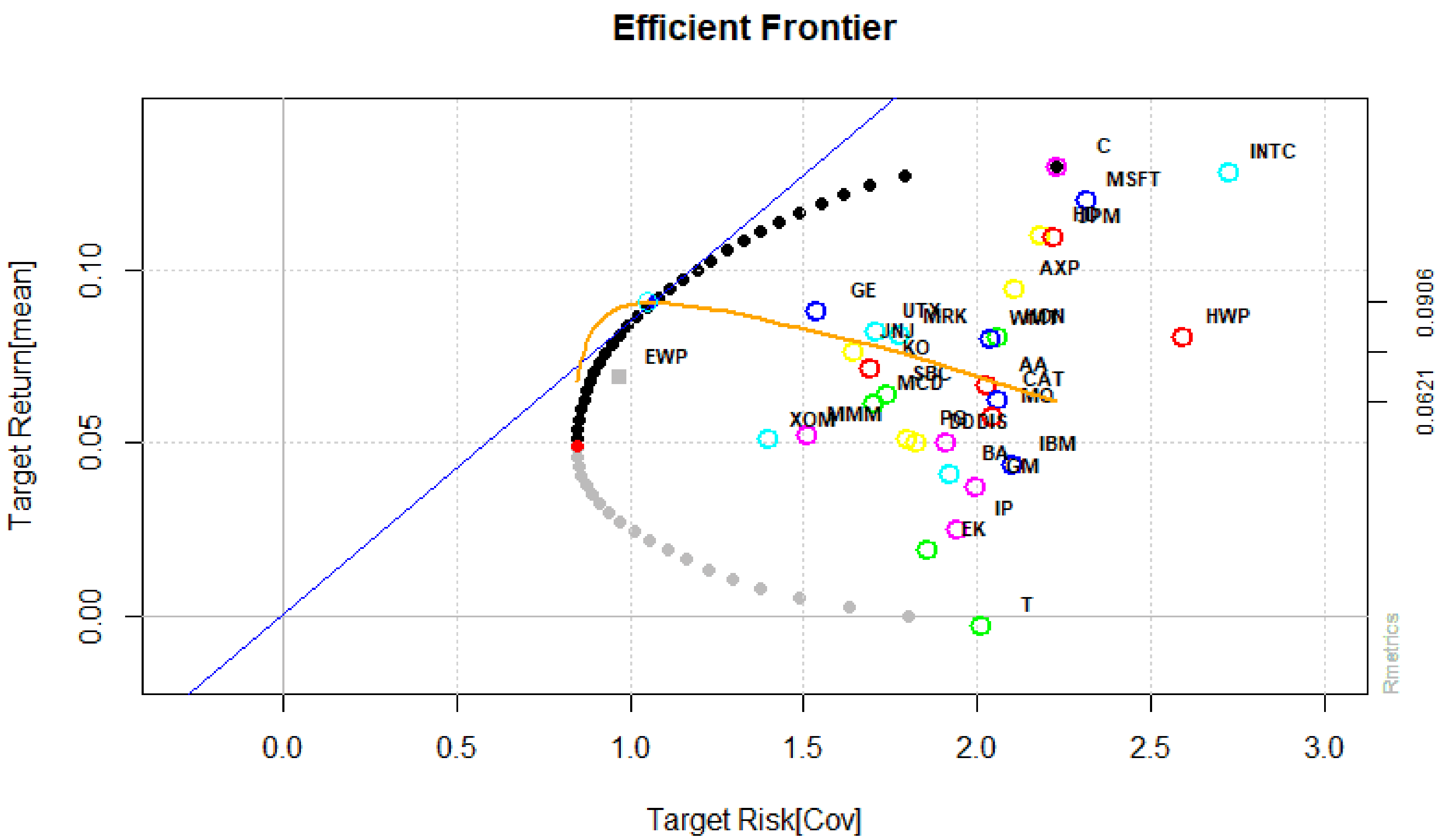
October 19, 2021

MY OBJECTIVES

1. Demonstrate that tremendous analytical value can be created using free open-source tools.
2. Show how powerful R is for financial analysis and visualization.
3. Demonstrate how data science is creating edge for those applying it to certain retailers.
   - Discuss the ethics of "Have and Have Not" data sets.
   - Where is Mr. Market? How is this being missed?

MEAN VARIANCE PORTFOLIO

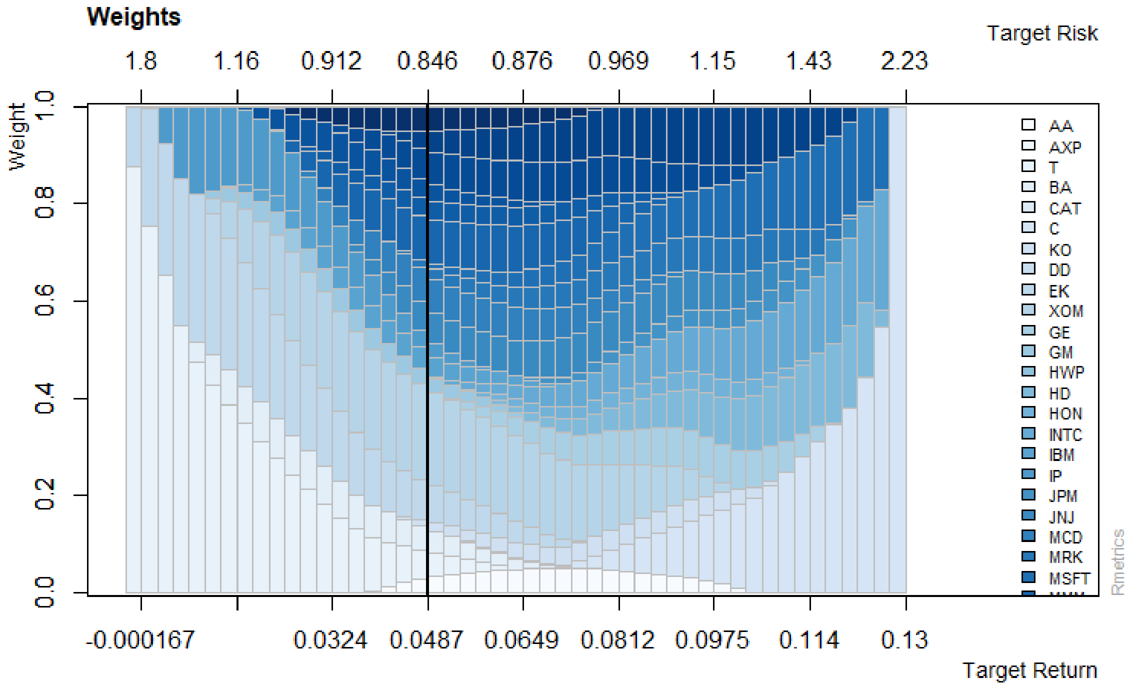
Mixing uncorrelated stocks gives you the benefits of their returns and hopefully reduces risk by exploiting opposing moves in stocks (Stock A goes up when Stock B goes down.)



Graphic from: https://miltonfmr.com/the-complete-guide-to-portfolio-optimization-in-r-part2/

WEIGHTS OF STOCK IN PORTFOLIO

Moving along the boundary (frontier) of the curve we see the number of stocks and their weights changing. (Note the extreme end points are dominated by: Low Risk, Low return stock at left, High Risk, High Return stock at right.



fPortfolio R library; Graphic from: https://miltonfmr.com/the-complete-guide-to-portfolio-optimization-in-r-part2/

LET'S TRY THIS AT HOME.

Using R. ...for free ...with free stock data

GET STOCK DATA, PLOT PERFORMANCE (7 LINES)

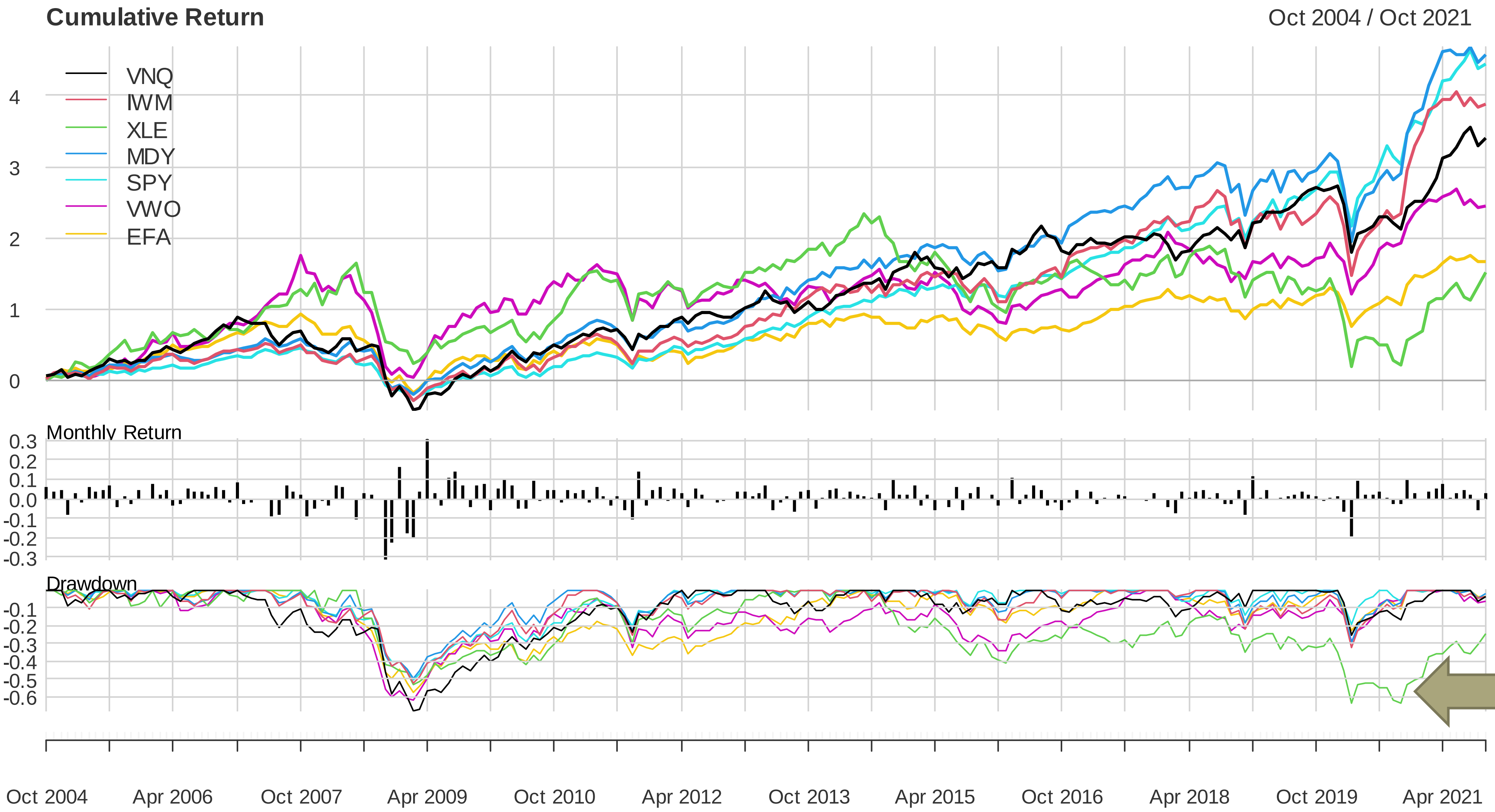
```
# Russel 2000; Emerging Market; SP500; developed Market;
# MidCap; Energy; Real Estate
TICS_ETFS <- c("IWM","VWO","SPY","EFA","MDY","XLE","VNQ")

e_ETF <- new.env()                                     # new environment
getSymbols(TICS_ETFS,from="1990-01-01",env=e_ETF) # fetch stocks into env

# Fetch price from env; convert to monthly; select "Adjusted Price"
tmp <- do.call(merge,lapply(eapply(e_ETF,to.monthly),Ad))["199412::"]
colnames(tmp) <- names(lapply(eapply(e_ETF,to.monthly),Ad))

RETS <- round(ROC(tmp,type="discrete"),4)
charts.PerformanceSummary(RETS,cex.legend=1.0, main="ETF Performance")
```

ETF Performance



Drawdown: The maximum loss relative to most recent high-water mark (1.0)

CREATE EQUAL WEIGHT BENCHMARK (2 LINES)

```
R <- TRET["2018/"]
funds <- colnames(R)

# Create a vector of equal weights
equal_weights <- rep(1 / ncol(R), ncol(R))  # Creates vector (1/n, 1/n, 1/n)

# Compute the benchmark returns
ref_benchmark <- Return.portfolio(R = R, weights = equal_weights,
                                  rebalance_on = "quarters")
colnames(ref_benchmark) <- "benchmark"      # Name the return series
```

CREATE PORTFOLIO CONSTRAINTS (8 LINES)

```
## optimized portfolio
port1 <- portfolio.spec(assets = funds)

port1 <- add.objective(portfolio = port1, type = "return", name = "mean")
port1 <- add.objective(portfolio = port1, type = "risk", name = "StdDev", multiplier=0)

port1 <- add.constraint(portfolio = port1, type = "risk", risk_target=0)
port1 <- add.constraint(portfolio = port1, type = "position_limit", max_pos_long=min(5,ncol(R)))
port1 <- add.constraint(portfolio = port1, type = "weight_sum", min_sum=0.97, max_sum=1.01)
port1 <- add.constraint(portfolio = port1, type = "long_only")

# Hold at least 5% but no more than 50%
c_min = rep(0.05, ncol(R))
c_max = rep(0.50, ncol(R))
port1 <- add.constraint(portfolio = port1, type = "box", enabled = TRUE, min=c_min, max=c_max)
```

PERFORM MEAN-VARIANCE OPTIMIZATION (1 MAGICAL, BUT FUSSY LINE)

```
# Rebalanced portfolio
opt1 <- optimize.portfolio.rebalancing(R=R, portfolio = port1, optimize_method = "DEoptim",
itermax=200,
                                      search_size = 2000, trace = TRUE, rebalance_on = "months",
                                      training_period = 60, rolling_window = 60)

# running it with "DEOptim" instead of "DEoptim" leads to silent failures "R, I love you, but we need
to talk"

[1] 0.058 0.050 0.498 0.074 0.182 0.066 0.056
Iteration: 1 bestvalit: -0.001434 bestmemit:    0.050000    0.128000    0.498000    0.070000    0.062000    0.132000
0.052000
Iteration: 2 bestvalit: -0.001434 bestmemit:    0.050000    0.128000    0.498000    0.070000    0.062000    0.132000
0.052000
Iteration: 3 bestvalit: -0.001443 bestmemit:    0.060000    0.080000    0.494000    0.094000    0.106000    0.080000
0.066000
Iteration: 4 bestvalit: -0.001443 bestmemit:    0.060000    0.080000    0.494000    0.094000    0.106000    0.080000
0.066000
...

# Chart Weights
chart.Weights(opt1)
```

Intuition regarding changes.

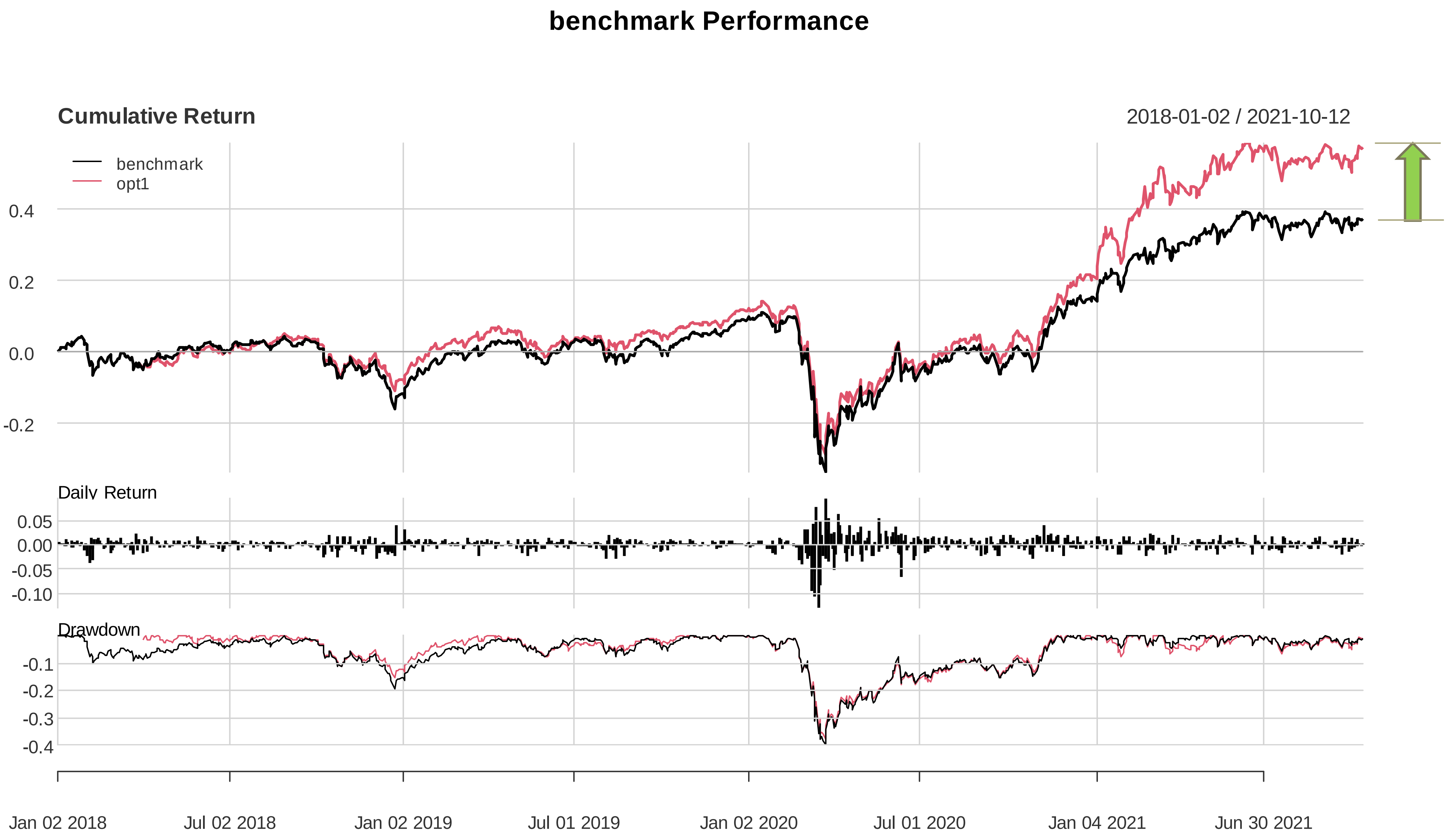
COMPUTE PORTFOLIO RETURN, COMPARE PERFORMANCE (3 LINES)

```
# Compute the portfolio returns
returns_opt1 <- Return.portfolio(R = R, weights = extractWeights(opt1))
colnames(returns_opt1) <- "opt1"

###-------------------------------------------
# Combine the returns
ret <- cbind(ref_benchmark, returns_opt1)
W
# Compute annualized returns
table.AnnualizedReturns(R = ret)
 benchmark   opt1
Annualized Return          0.0878 0.1462
Annualized Std Dev         0.2255 0.2202
Annualized Sharpe (Rf=0%)  0.3893 0.6641
# Chart the performance summary
charts.PerformanceSummary(R = ret)
```

Nice!

We improved annualized return by 66%.
Slightly reduced risk
Increased Sharpe ratio by 70.6%
Cost: $0, pennies of electricity!



Better Performance, Less Risk to get it.

Win-Win in this case!

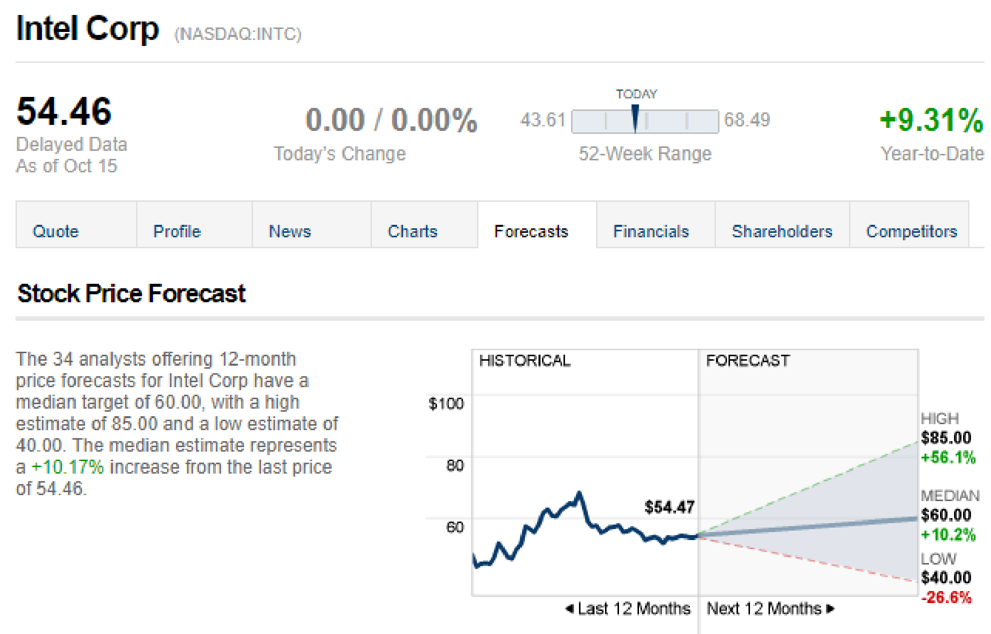
Lost less during most downturns. (Red line above Black line)

WHICH STOCK TO SELL? OR WRITE CALLS AGAINST?

You'd like to sell or (write calls against) a stock that topping out, but you want a second opinion. Let's use R to get several opinions.

CNN FINANCIAL ANALYST PAGE

Trying to remember the analyst estimates on more than three stocks is tedious. I want to compare ~50.



BETTER PERSPECTIVE

Using Python (or R) to scrape the data and then render as a scatter plot:

Vertical Axis: Fraction of highest analyst target 1-year price.

Horizontal Axis: Fraction of mean analyst target 1-year price.

Vertical line = 1.0 (stocks to the right are above the analysts mean target.

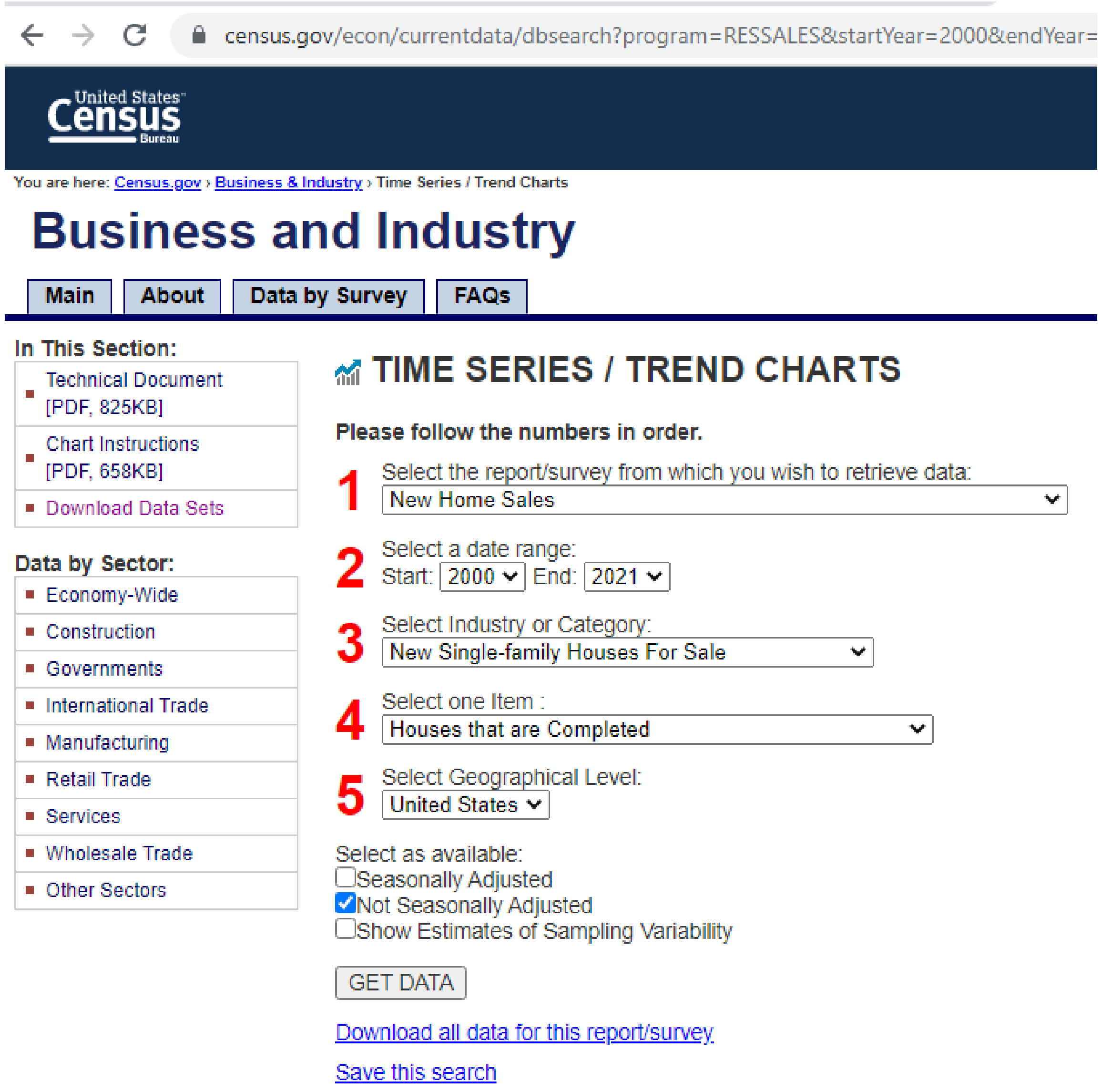
LET'S LOOK AT HOUSING TRENDS AND DETRENDS

Seasonally adjusted is a term you hear often. Let's take a look at housing sales of complete homes using R.

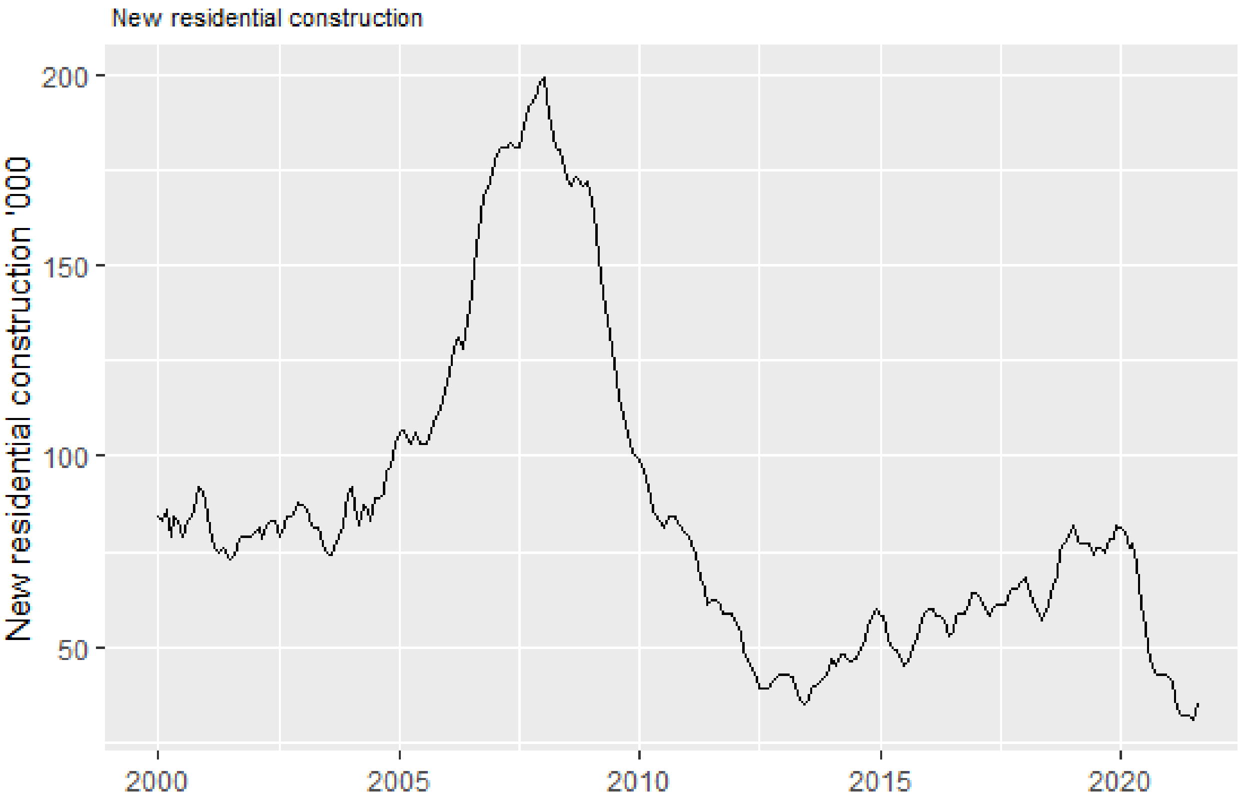
TIME SERIES DATA FROM U.S. CENSUS

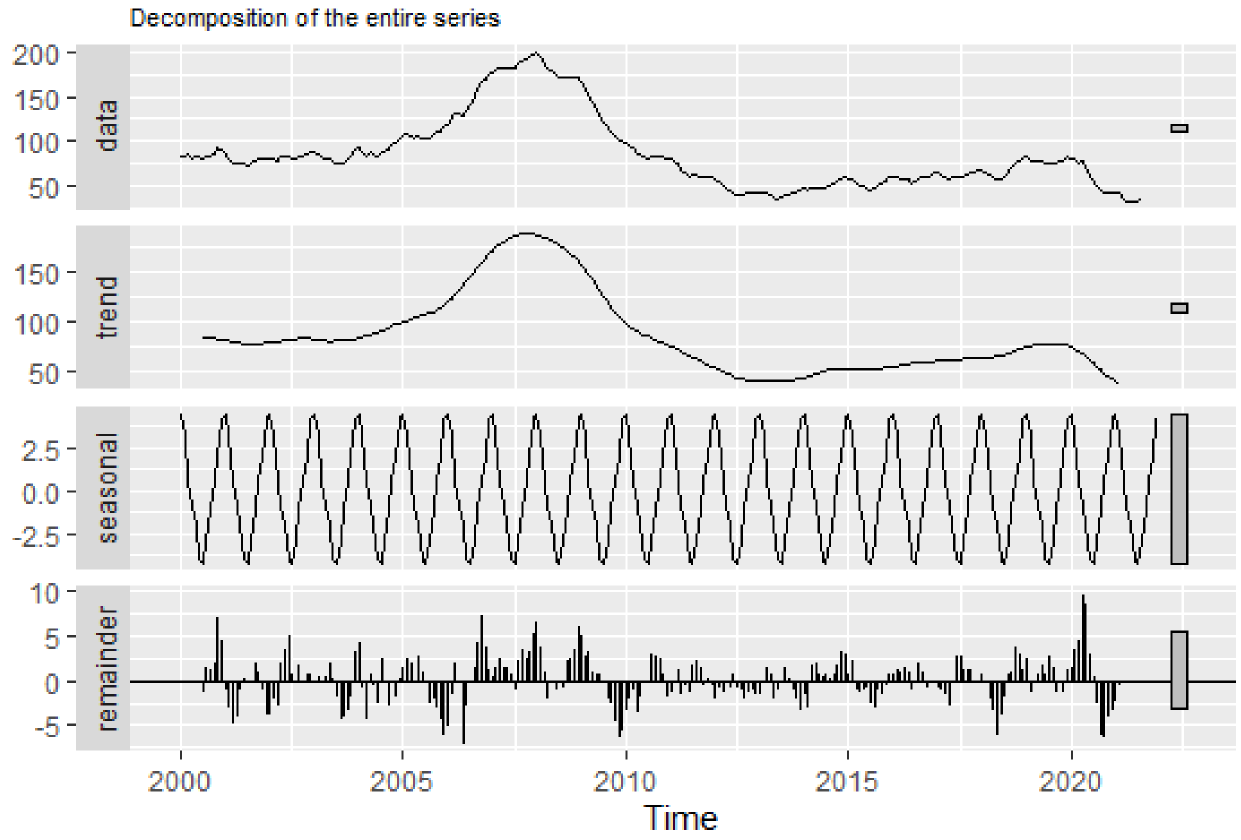
Free data! (actually, you already paid for it)

https://www.census.gov/econ/currentdata/



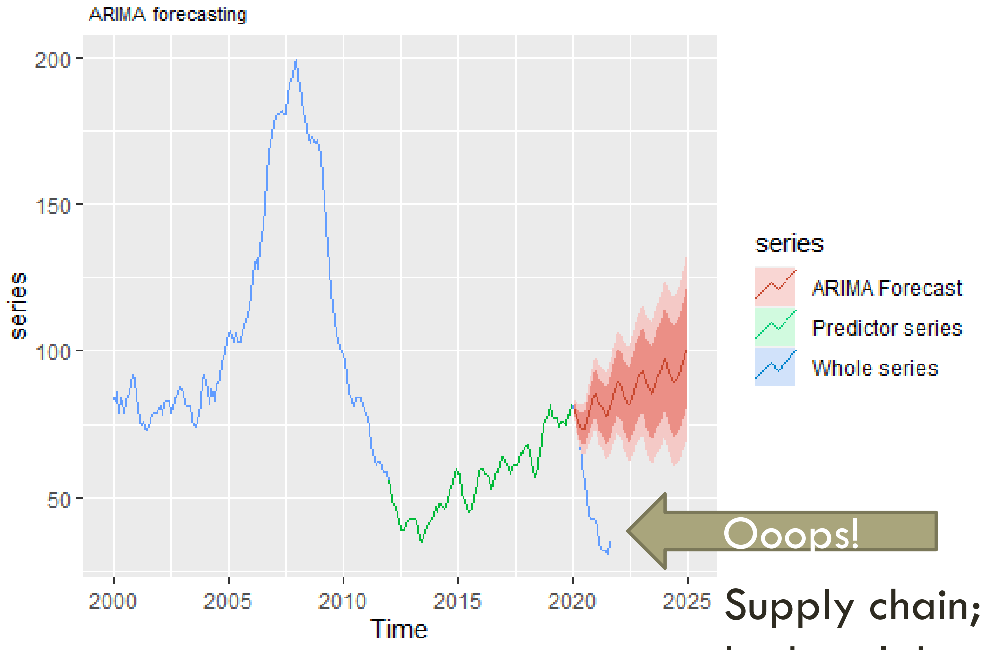
20 YEARS OF HOUSING DATA





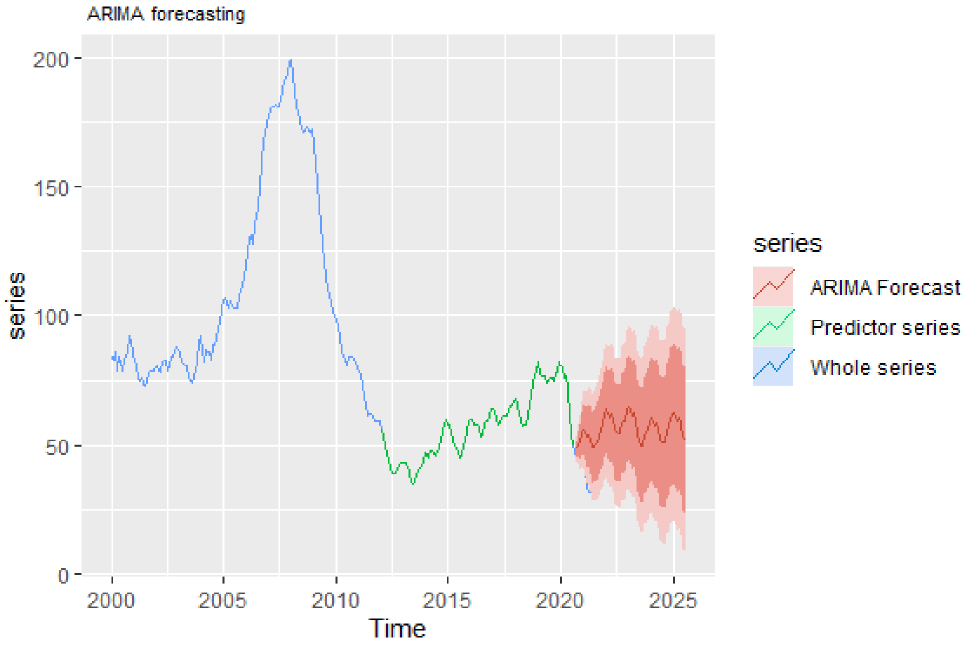
WHAT A DIFFERENCE 8 MONTHS & SUPPLY CHAIN ISSUES MAKE!

Using Prediction Window 2020-Jan



Supply chain; lumber; labor

Using Prediction Window 2020-Aug!!



R CODE: TIME SERIES PLOT & DECOMPOSITION

```r
library(fpp2)
library(ggplot2)
# import data
df = read.csv("./HousingCons.csv", header=F)
# check out first few rows
head(df)

# convert  values into a time series object
series = ts(df[,c(2)], start = 2000, frequency = 12)
# now check out the series
print(series)

# plot the series
autoplot(series) +
  xlab(" ") +
  ylab("New residential construction '000") +
  ggtitle(" New residential construction") +
  theme(plot.title = element_text(size=8))
# decomposition
options(repr.plot.width = 6, repr.plot.height = 3)
autoplot(decompose(series)) + ggtitle("Decomposition of the entire series")+
  theme(plot.title = element_text(size=8))
```

R CODE: TIME SERIES ARIMA FORECAST

```r
# model
forecast_arima = auto.arima(predictor_series, seasonal=TRUE, stepwise = FALSE,
                            approximation = FALSE)
forecast_arima = forecast(forecast_arima, h=60)
# plot
autoplot(series, series=" Whole series") +
  autolayer(predictor_series, series=" Predictor series") +
  autolayer(forecast_arima, series=" ARIMA Forecast") +
  ggtitle(" ARIMA forecasting") +
  theme(plot.title = element_text(size=8))
```

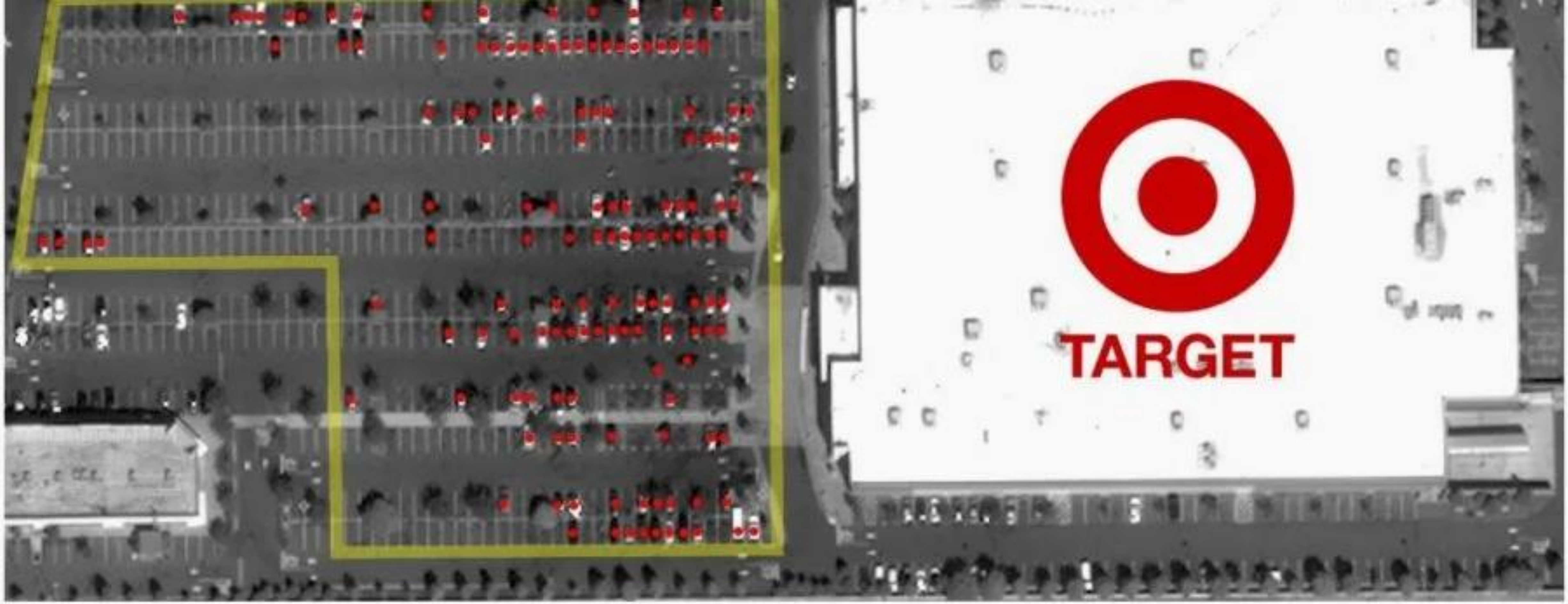
Great reference: https://otexts.com/fpp2/tspatterns.html

Portions from: https://towardsdatascience.com/applied-time-series-forecasting-residential-housing-in-the-us-f8ab68e63f94

DATA SCIENCE PROVIDING "EDGE"

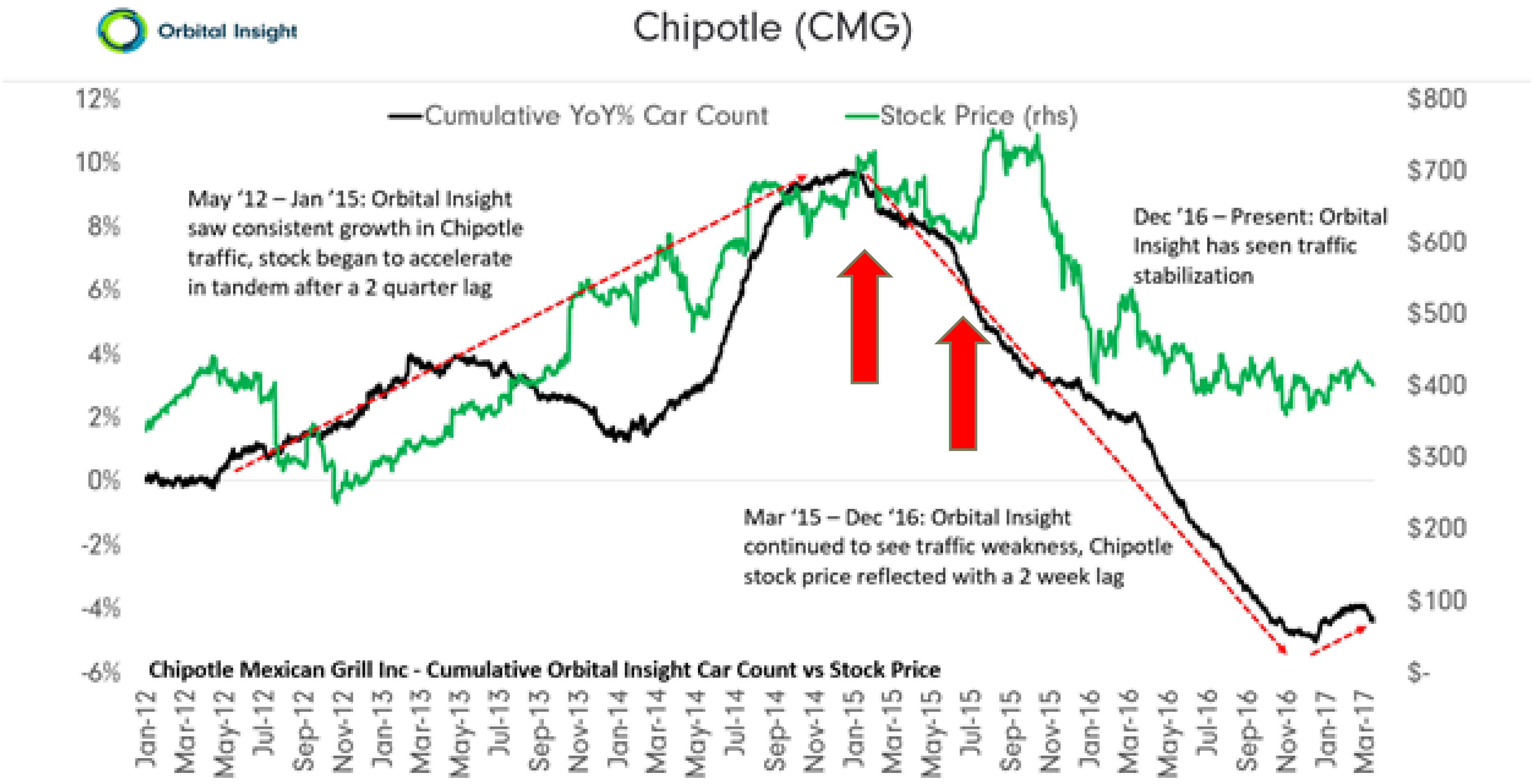
Alternative data predicting stock prices is largely ignored by the market (as of 2018)

On the Capital Market Consequences of Alternative Data: Evidence from Outer Space



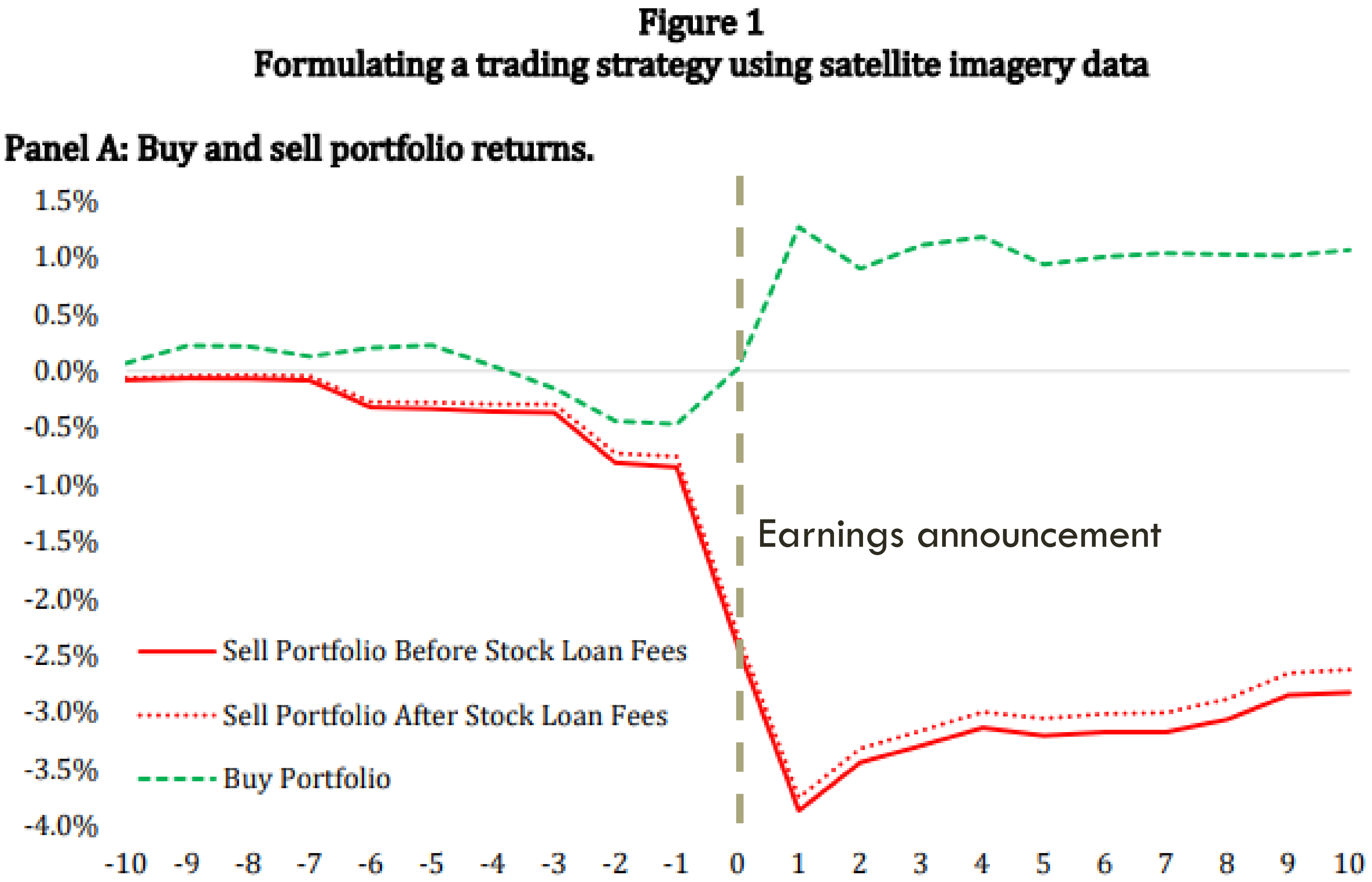
SATELLITE DATA IS PREDICTOR OF SAME STORE SALES

This data is available before quarterly earnings announcements, yet the market doesn't appear to reprice.



CHIPOTLE'S TRAFFIC DROP VS STOCK PRICE (E.COLI)

https://medium.com/from-the-macroscope/case-study-how-chipotle-car-counts-over-the-last-5-years-correlate-with-stock-prices-264e872c254f



BUY (SELL) TOP (BOTTOM) QUARTILE OF FILL RATE

Although stock analysts first proposed using parking lot data, as of 2018 they rarely adjusted targets based on the data prior to earnings announcements leading to an exploitable return. (Is this ethical?)

Table 3
Formulating a trading strategy using satellite imagery data

Panel A: Portfolio performance before stock loan fees.

	Portfolio returns before stock loan fees		
	Raw Returns	Market Adjusted	Factor Adjusted
Sell Portfolio	-2.82%***	-3.01%***	-3.10%***
	(-2.90)	(-3.13)	(-3.25)
Buy Portfolio	1.78%**	1.63%**	1.66%**
	(2.38)	(2.17)	(2.22)
Buy-minus-sell	4.60%***	4.64%***	4.76%***
	(3.75)	(3.80)	(3.93)

BUY (SHORT) STOCK WITH INCREASING (DECREASING) FILL RATES IN PARKING LOTS: 4.60% – 4.76%

This table reports returns from a trading strategy that buys (short sells) retailers with same-store parking lot fill rate growth in the top (bottom) quartile of the cross-sectional distribution of ΔFLRTiq.

PARTING THOUGHTS…

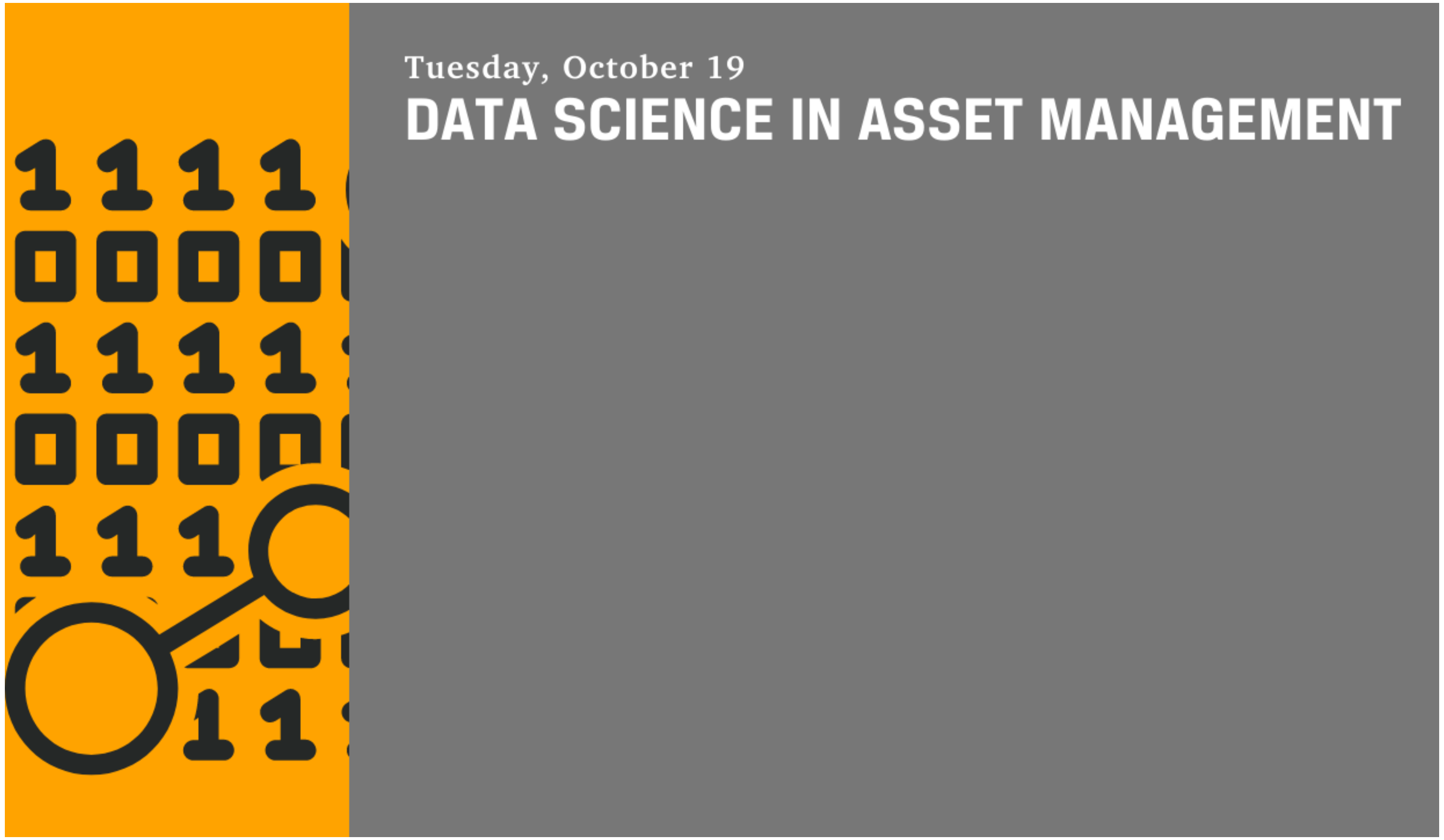
: Patatoukas: co-author of study

In a certain sense, buying and selling stocks based on satellite data resembles insider trading. One reason insider trading is illegal is that it benefits those with superior information and deceives outsiders who lack such an advantage. Sure, the number of cars in a parking lot is technically public information, but as a practical matter, few investors have the resources to take advantage of it. “Isn’t trading based on satellite information similar to trading based on material nonpublic information?”

Counter Arguments

CEO of RS Metrics (satellite data): Expensive now but will soon be cheaper and then everyone will have it.

The Market dictates that the value of the cheaper widespread information will fall with price.



WELCOME TO PART 2: PROFESSIONAL PANEL SESSION