

# THE US INFLATION PHENOMENON



**AUTHOR** | Rand Sobczak Jr.

**DATE** | 21 September 2021

# Table of contents

01

Problem Identification

02

Generated Deliverables

03

Data Pre-Processing

04

Model Description

05

Model Findings

06

Next Steps

# 01



## Problem Identification

Developing a model to explain & understand  
the phenomenon of US Inflation



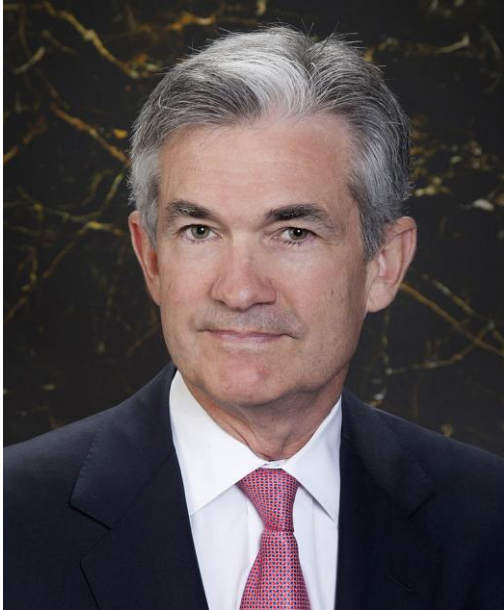
# What is Inflation?

**Inflation** is the **decline of purchasing power** of a given currency over time. **Deflation is the inverse**



# Inflation is important

but it's a **highly debated** phenomenon in economics. Many economists maintain that **moderate** inflation **levels** are needed to **drive consumption**, assuming that higher levels of **spending are crucial** for **economic growth**



# Inflation is important

but it's a highly debated phenomenon in economics. Many economists maintain that moderate inflation levels are needed to drive consumption, assuming that higher levels of spending are crucial for economic growth

& **stabilizing Inflation** is one of three objectives of the **Federal Reserve** who's decisions move the global **financials markets**

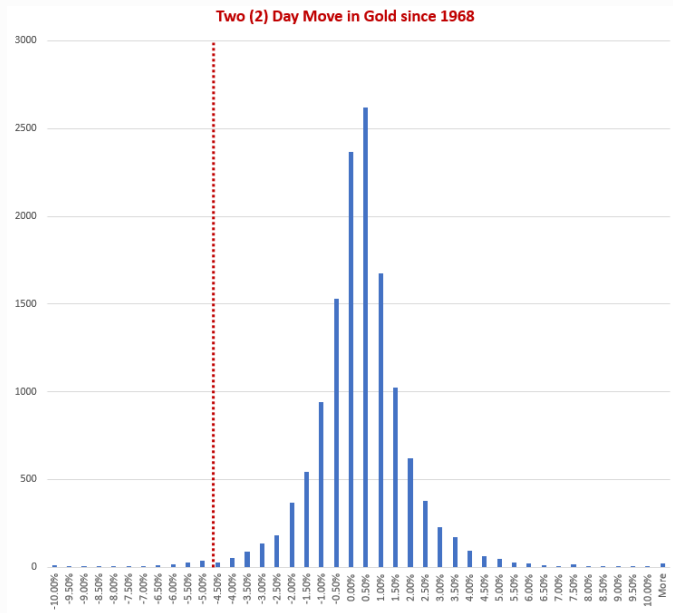


# Inflation is important

but it's a highly debated phenomenon in economics. Many economists maintain that moderate inflation levels are needed to drive consumption, assuming that higher levels of spending are crucial for economic growth

& stabilizing it is one of three objectives of the Federal Reserve who's decisions move the global financial markets

**Gold**, for example, saw **2 day drop of 4.67%** the day of & after the Fed mentioned tapering; i.e., **raising the Fed Rate in response to Inflation**



# Inflation is important

but it's a highly debated phenomenon in economics. Many economists maintain that moderate inflation levels are needed to drive consumption, assuming that higher levels of spending are crucial for economic growth

& stabilizing it is one of three objectives of the Federal Reserve who's decisions move the global financials markets

Gold, for example, saw 2 day drop of 4.67% the day of & after the Fed mentioned tapering; i.e., raising the Fed Rate in response to Inflation

**Math language, a 2+ standard deviated move**





The purpose & goal of this Data Science project is to

**build a model to**  
**explain & understand**  
**the phenomenon of**  
**US Inflation**

# 02



## Generated Deliverables

The power of API's

# Generated Deliverables



## Quandl

*Quandl is a marketplace for financial, economic and alternative data*



## Investing.com

*A financial platform & news website; one of the top 3 financial websites in the world*



## FRED

*Federal Reserve Economic Data (FRED) a database maintained by the Research division of the Federal Reserve Bank of St. Louis*

# Problem Identification

( cont. )

I **shortlisted 19 variables** to determine their influence on Inflation

Items	Reported	API	API Source		Comments
Inflation	Monthly	Quandl	U.S. Bureau of Labor Statistics		The target variable
Wages CPI	Monthly	FRED	U.S. Bureau of Labor Statistics		A component of the target variable
WTI	Daily	Quandl	CME	West Texas Intermediate - One of many commodities	
Heating Oil	Daily	Investpy	Investing.com		One of many commodities
Copper	Daily	Investpy	Investing.com		One of many commodities
Sugar	Daily	Investpy	Investing.com		One of many commodities
Natural Gas	Daily	Investpy	Investing.com		One of many commodities
Cattle	Daily	Investpy	Investing.com		One of many commodities
Lean Hogs	Daily	Investpy	Investing.com		One of many commodities
Soybeans	Daily	Investpy	Investing.com		One of many commodities
Lumber	Daily	Investpy	Investing.com		One of many commodities
Capacity Utilization	Monthly	FRED	Board of Governors of the Federal Reserve		The % of resources used by corporations
Corn	Daily	Investpy	Investing.com		One of many commodities
M2 Velocity	Quarterly	FRED	Federal Reserve Bank of St. Louis	Movement of money; state of the economy proxy	
GDP	Quarterly	FRED	U.S. Bureau of Economic Analysis		A proxy for the state of the economy
Wheat	Daily	Investpy	Investing.com		One of many commodities
PMI	Monthly	Quandl	Institute of Supply Management	Manufacturing PMI - A proxy for the economy	
USD Index	Daily	Quandl	Intercontinental Exchange Inc	( DXY ) Proxy for potentially importing inflation	
Unemployment Rate	Monthly	Quandl	U.S. Bureau of Labor Statistics		A proxy for the state of the economy
Initial Jobless Claims	Weekly	Quandl	U.S. Employment and Training Administration		A proxy for the state of the economy

## Target Variable

Commodities  
Economic Data

I **Target variable** | What we seek to understand

Items	Reported	API	API Source	Comments
Inflation	Monthly	Quandl	U.S. Bureau of Labor Statistics	The target variable

Target Variable

## Commodities

Economic Data

### I **Commodities** | Where Inflation typically shows itself

Items	Reported	API	API Source	Comments
WTI	Daily	Quandl	CME	West Texas Intermediate - One of many commodities
Heating Oil	Daily	Investpy	Investing.com	One of many commodities
Copper	Daily	Investpy	Investing.com	One of many commodities
Sugar	Daily	Investpy	Investing.com	One of many commodities
Natural Gas	Daily	Investpy	Investing.com	One of many commodities
Cattle	Daily	Investpy	Investing.com	One of many commodities
Lean Hogs	Daily	Investpy	Investing.com	One of many commodities
Soybeans	Daily	Investpy	Investing.com	One of many commodities
Lumber	Daily	Investpy	Investing.com	One of many commodities
Corn	Daily	Investpy	Investing.com	One of many commodities
Wheat	Daily	Investpy	Investing.com	One of many commodities

## Economic Data

### I **Economic Data** | Variables to determine the health of the economy

Items	Reported	API	API Source	Comments
Wages CPI	Monthly	FRED	U.S. Bureau of Labor Statistics	A component of the target variable
Capacity Utilization	Monthly	FRED	Board of Governors of the Federal Reserve	The % of resources used by corporations
M2 Velocity	Quarterly	FRED	Federal Reserve Bank of St. Louis	Movement of money; state of the economy proxy
GDP	Quarterly	FRED	U.S. Bureau of Economic Analysis	A proxy for the state of the economy
PMI	Monthly	Quandl	Institute of Supply Management	Manufacturing PMI - A proxy for the economy
USD Index	Daily	Quandl	Intercontinental Exchange Inc	( DXY ) Proxy for potentially importing inflation
Unemployment Rate	Monthly	Quandl	U.S. Bureau of Labor Statistics	A proxy for the state of the economy
Initial Jobless Claims	Weekly	Quandl	U.S. Employment and Training Administration	A proxy for the state of the economy

# Generated Deliverables

( cont. )



## Source Code

*This can be found at my GitHub account referenced at the end*



## Research Report

*Also can be found at my GitHub account referenced at the end*



## Presentation Report

*This one...*



# 03



## Data Pre-Processing

Split it up...

# Data

## Pre-Processing

Data Cleaning

## Data Frames should talk to each other

- After pulling, the data frame was **composed of variables with different lengths**

```
<class 'pandas.core.frame.DataFrame'>
DatetimeIndex: 14302 entries, 1946-01-01 to 2021-09-03
Data columns (total 19 columns):
#   Column                                Non-Null Count  Dtype
---  -
0   Wage CPI                             14293 non-null  float64
1   WTI                                   12088 non-null  float64
2   Heating Oil                           13087 non-null  float64
3   Copper                                10440 non-null  float64
4   Sugar                                 13087 non-null  float64
5   Natural Gas                           9915 non-null   float64
6   Cattle                                13084 non-null  float64
7   Lean Hogs                             13089 non-null  float64
8   Soybeans                              9999 non-null   float64
9   Lumber                                13089 non-null  float64
10  Capacity Utilization                  14033 non-null  float64
11  Corn                                  13086 non-null  float64
12  M2 Velocity                           14151 non-null  float64
13  GDP                                   14295 non-null  float64
14  Wheat                                 10001 non-null  float64
15  PMI                                   14281 non-null  float64
16  USD Index                             11273 non-null  float64
17  Unemployment Rate                     14281 non-null  float64
18  Initial Jobless Claims                14030 non-null  float64
dtypes: float64(19)
memory usage: 2.2 MB
```

# Data Pre-Processing

## Data Cleaning

## Data Frames should talk to each other

- After pulling, the data frame was composed of variables with different lengths
  - Natural Gas being the constraint**
  - Used forward fill**

```
<class 'pandas.core.frame.DataFrame'>
DatetimeIndex: 9752 entries, 1991-04-18 to 2021-09-03
Data columns (total 19 columns):
#   Column              Non-Null Count  Dtype
---  -
0   Wage CPI            9752 non-null  float64
1   WTI                 9752 non-null  float64
2   Heating Oil         9752 non-null  float64
3   Copper              9752 non-null  float64
4   Sugar               9752 non-null  float64
5   Natural Gas         9752 non-null  float64
6   Cattle              9752 non-null  float64
7   Lean Hogs           9752 non-null  float64
8   Soybeans            9752 non-null  float64
9   Lumber              9752 non-null  float64
10  Capacity Utilization 9752 non-null  float64
11  Corn                9752 non-null  float64
12  M2 Velocity         9752 non-null  float64
13  GDP                 9752 non-null  float64
14  Wheat               9752 non-null  float64
15  PMI                 9752 non-null  float64
16  USD Index           9752 non-null  float64
17  Unemployment Rate   9752 non-null  float64
18  Initial Jobless Claims 9752 non-null  float64
dtypes: float64(19)
memory usage: 1.5 MB
```

# Data Pre-Processing

Data Cleaning ( cont. )

## Data Frames should talk to each other ( cont. )

- Different lengths
- **Cut the data to April 1991**

```
<class 'pandas.core.frame.DataFrame'>
DatetimeIndex: 9752 entries, 1991-04-18 to 2021-09-03
Data columns (total 19 columns):
#   Column              Non-Null Count  Dtype
---  -
0   Wage CPI            9752 non-null   float64
1   WTI                  9752 non-null   float64
2   Heating Oil         9752 non-null   float64
3   Copper              9752 non-null   float64
4   Sugar               9752 non-null   float64
5   Natural Gas         9752 non-null   float64
6   Cattle              9752 non-null   float64
7   Lean Hogs           9752 non-null   float64
8   Soybeans            9752 non-null   float64
9   Lumber              9752 non-null   float64
10  Capacity Utilization 9752 non-null   float64
11  Corn                9752 non-null   float64
12  M2 Velocity         9752 non-null   float64
13  GDP                 9752 non-null   float64
14  Wheat               9752 non-null   float64
15  PMI                 9752 non-null   float64
16  USD Index           9752 non-null   float64
17  Unemployment Rate    9752 non-null   float64
18  Initial Jobless Claims 9752 non-null   float64
dtypes: float64(19)
memory usage: 1.5 MB
```

# Data Pre-Processing

Data Cleaning ( cont. )

## Data Frames should talk to each other ( cont. )

- Different lengths
- Cut the Data
- **Concatenated with Inflation**
  - Using a forward fill
  - Only 321 observations

```
<class 'pandas.core.frame.DataFrame'>
DatetimeIndex: 321 entries, 1991-04-30 to 2021-07-31
Data columns (total 20 columns):
#   Column                                Non-Null Count  Dtype
---  -
0   Inflation                             321 non-null    float64
1   Wage CPI                             321 non-null    float64
2   WTI                                   321 non-null    float64
3   Heating Oil                           321 non-null    float64
4   Copper                               321 non-null    float64
5   Sugar                                321 non-null    float64
6   Natural Gas                           321 non-null    float64
7   Cattle                               321 non-null    float64
8   Lean Hogs                             321 non-null    float64
9   Soybeans                              321 non-null    float64
10  Lumber                                321 non-null    float64
11  Capacity Utilization                  321 non-null    float64
12  Corn                                  321 non-null    float64
13  M2 Velocity                           321 non-null    float64
14  GDP                                   321 non-null    float64
15  Wheat                                 321 non-null    float64
16  PMI                                   321 non-null    float64
17  USD Index                             321 non-null    float64
18  Unemployment Rate                     321 non-null    float64
19  Initial Jobless Claims                321 non-null    float64
dtypes: float64(20)
memory usage: 52.7 KB
```

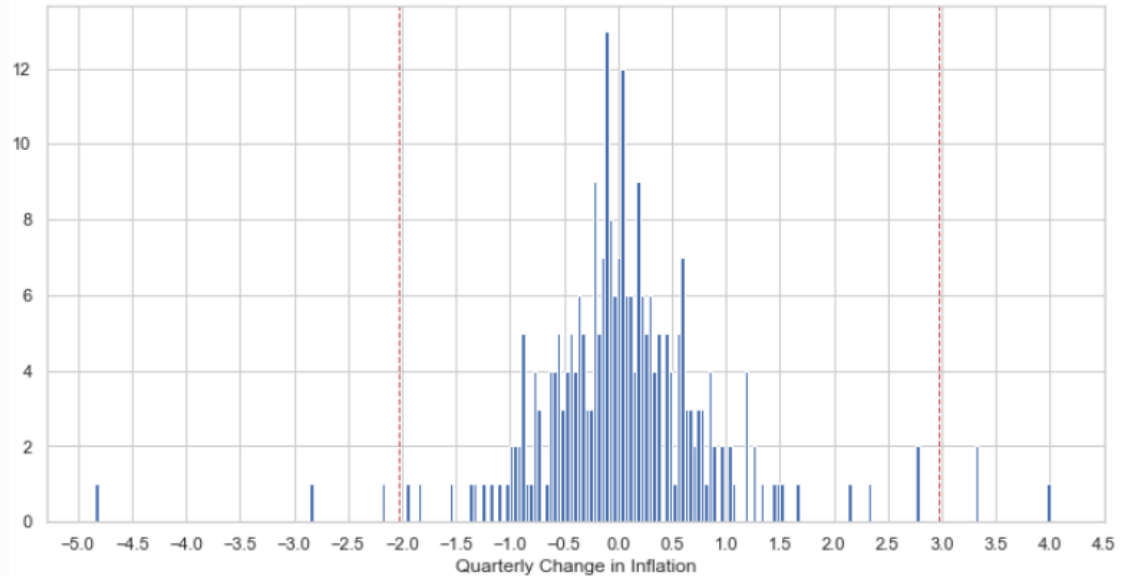
Items	Reported	API	API Source	Comments
Inflation	Monthly	Quandl	U.S. Bureau of Labor Statistics	The target variable

# Data Pre-Processing

Data Cleaning ( cont. )

## Winsorizing

- **Winsorizing** is the transformation of statistics by limiting extreme values in data **to reduce the effect of potential spurious outliers**

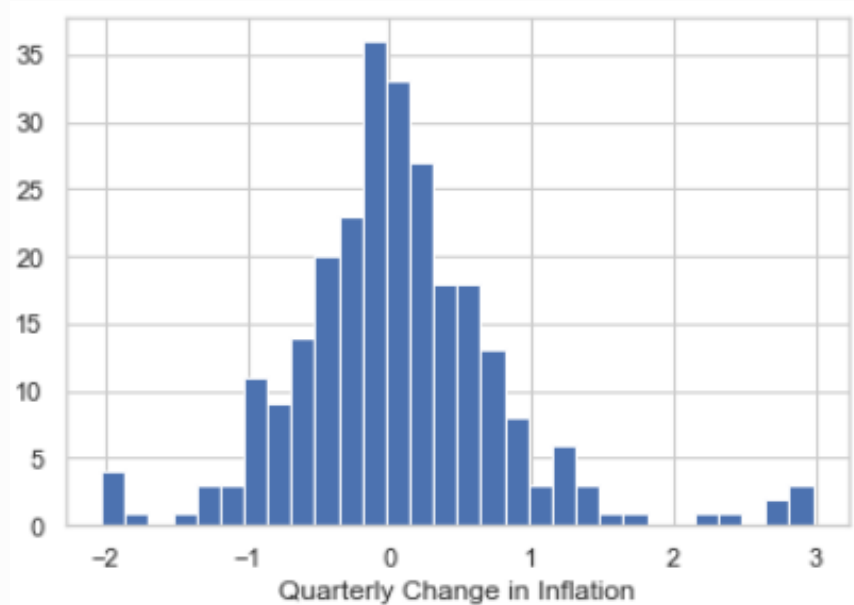


# Data Pre-Processing

Data Cleaning ( cont. )

## Winsorizing

- Winsorizing is the transformation of statistics by limiting extreme values in data to reduce the effect of potential spurious outliers
- **Inflation was Winsorized differently on each of the approaches ( described next )**

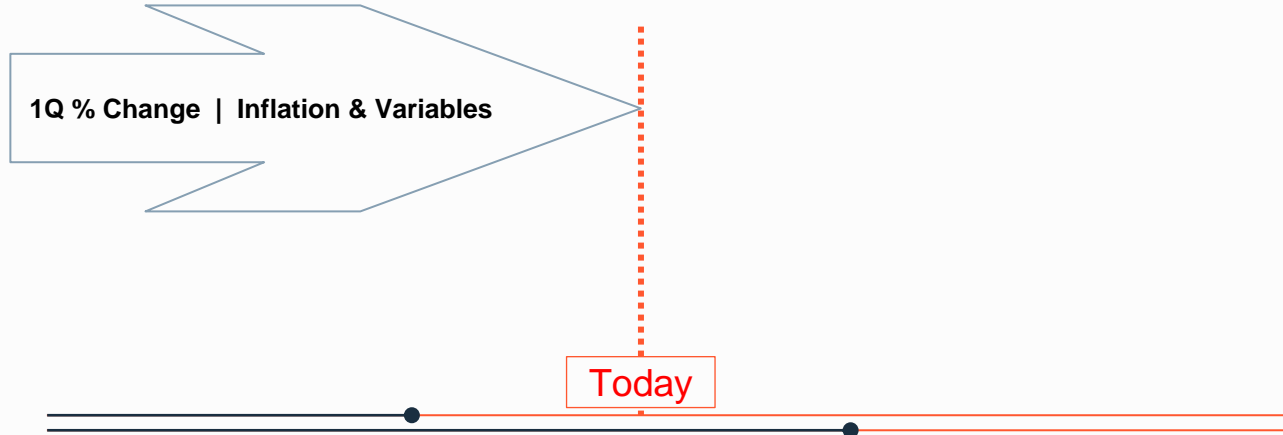


# Data Pre-Processing

Exploratory Data Analysis

## Investigating the Time Relationships

- **Quarter on Quarter ( for all )**
  - Compared a quarterly change on Variables & Inflation
- Month on Month ( for all )
- Quarter on Quarter for Variables ( past ) & Inflation ( forwards )
- Quarter on Quarter w/ Rolling Averages



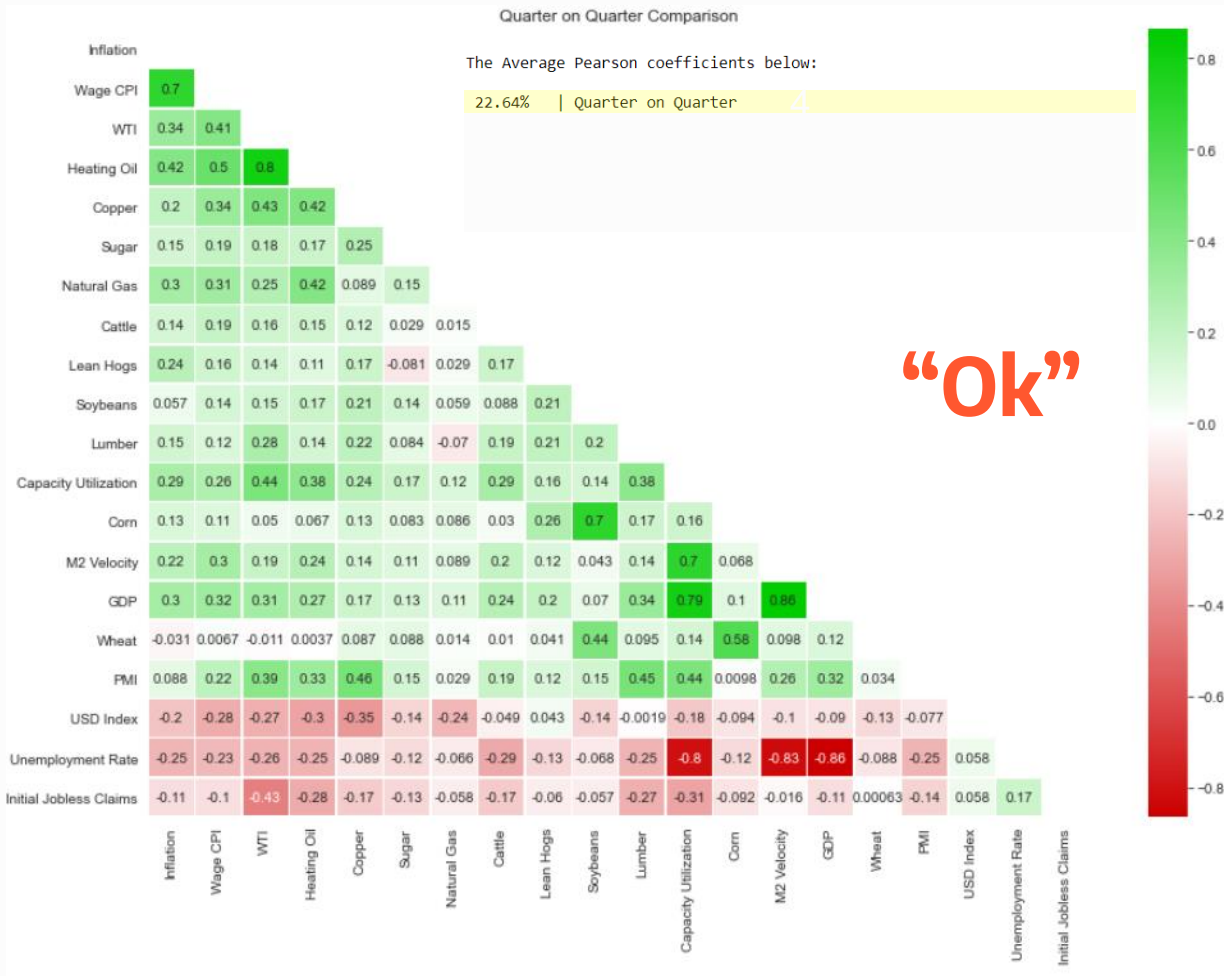


# Data Pre-Processing

Exploratory Data Analysis

Quarter on Quarter ( for all )

Feature Correlation Heat Maps with the  
Pearson correlation coefficients

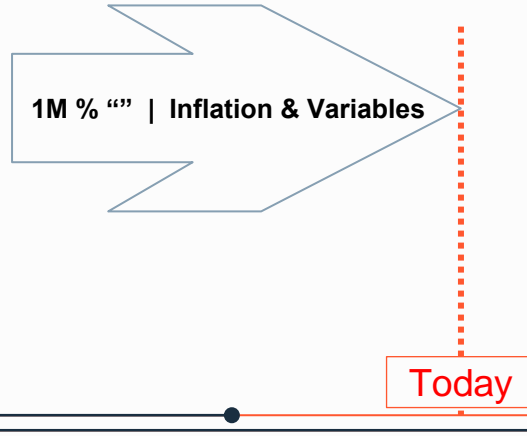


# Data Pre-Processing

Exploratory Data Analysis ( cont. )

## Investigating the Time Relationships ( cont. )

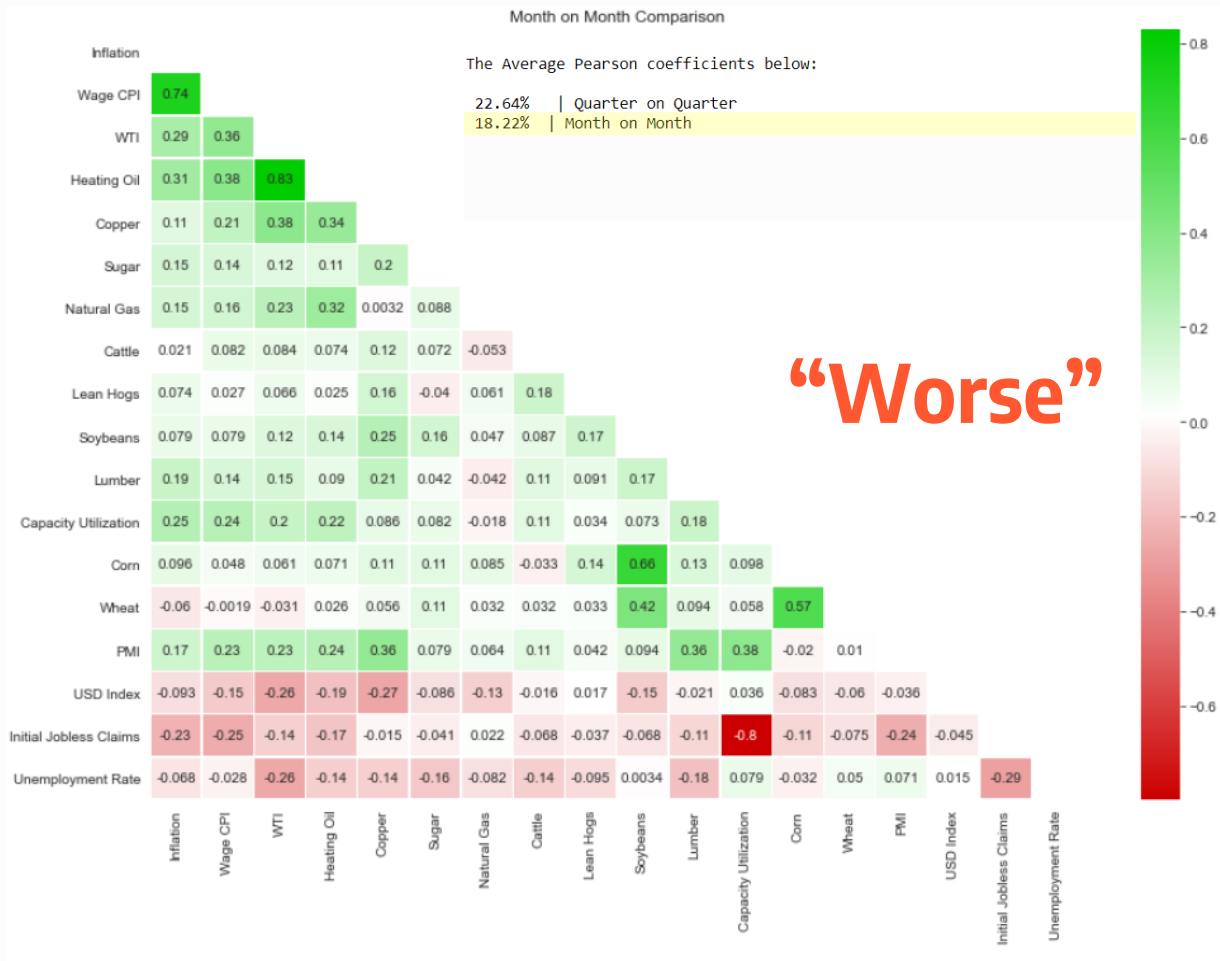
- Quarter on Quarter ( for all )
- **Month on Month ( for all )**
  - The same as the previous but looked at monthly change
- Quarter on Quarter for Variables ( past ) & Inflation ( forwards )
- Quarter on Quarter w/ Rolling Averages



# Data Pre-Processing

Exploratory Data Analysis

Month on Month ( for all )  
Feature Correlation Heat Maps with the  
Pearson correlation coefficients  
( cont. )



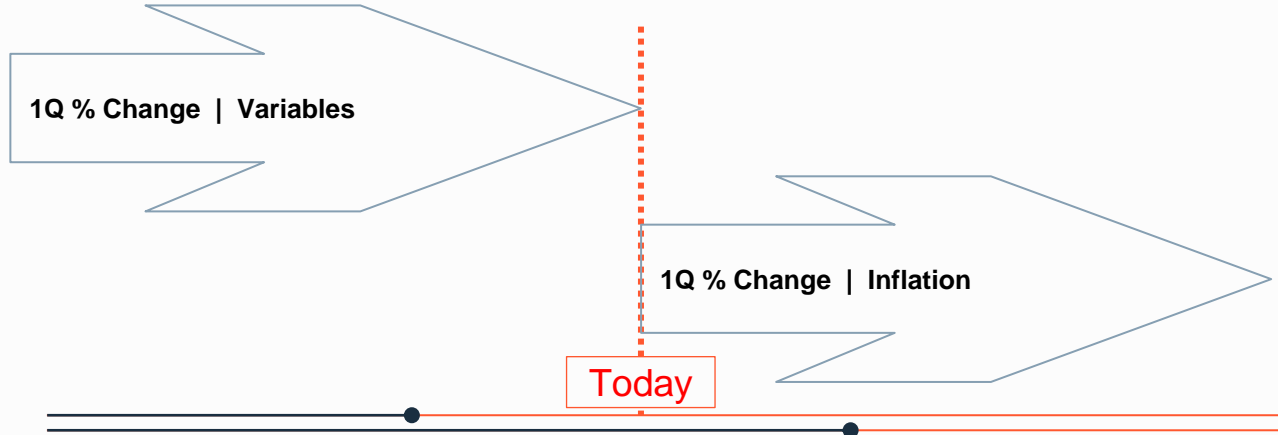
# Data Pre-Processing

Exploratory Data Analysis ( cont. )

## Data

## Investigating the Time Relationships ( cont. )

- Quarter on Quarter ( for all )
- Month on Month ( for all )
- **Q on Q for Variables ( past ) & Inflation ( forwards )**
  - Looked at a previous 1 quarter change for variables to a 1 quarter change in Inflation in the future
- Quarter on Quarter w/ Rolling Averages



# Data Pre-Processing

Exploratory Data Analysis

Q on Q for Variables ( past ) & Inflation ( forwards )

Feature Correlation Heat Maps with the Pearson correlation coefficients ( cont. )

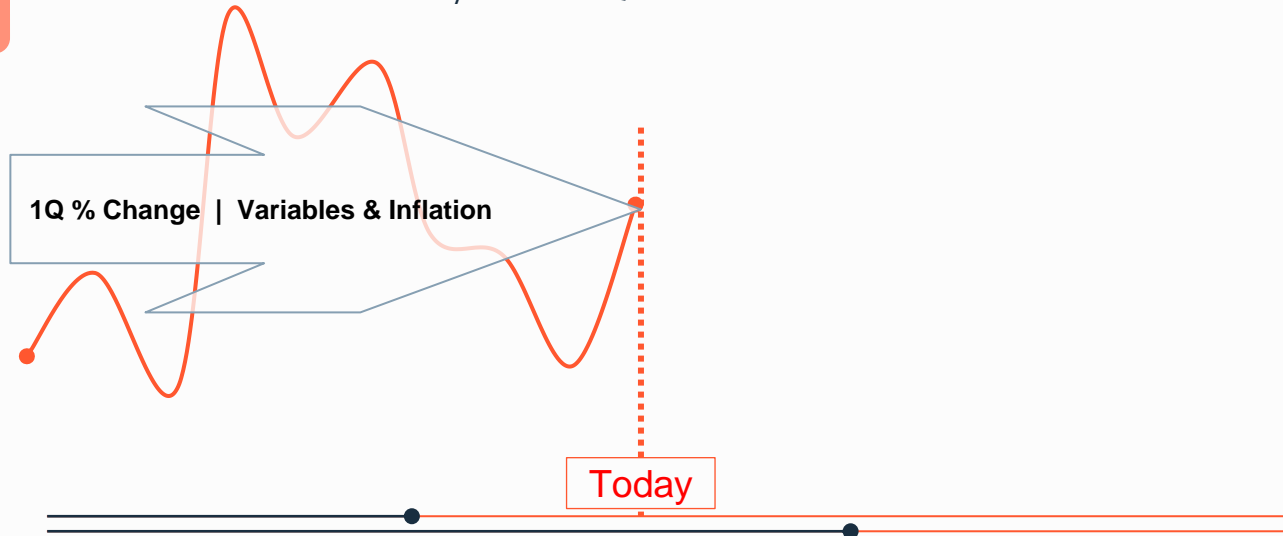


# Data Pre-Processing

Exploratory Data Analysis ( cont. )

## Investigating the Time Relationships ( cont. )

- Quarter on Quarter ( for all )
- Month on Month ( for all )
- Q on Q for Variables ( past ) & Inflation ( forwards )
- **Quarter on Quarter w/ Rolling Averages**
  - Similar to # 1 albeit used a rolling average for those that were reported more often than once a Quarter as a Variable “may have had” a bad week or day when the Quarter ended



# Data Pre-Processing

Exploratory Data Analysis

Quarter on Quarter w/ Rolling Averages  
Feature Correlation Heat Maps with the  
Pearson correlation coefficients  
( cont. )

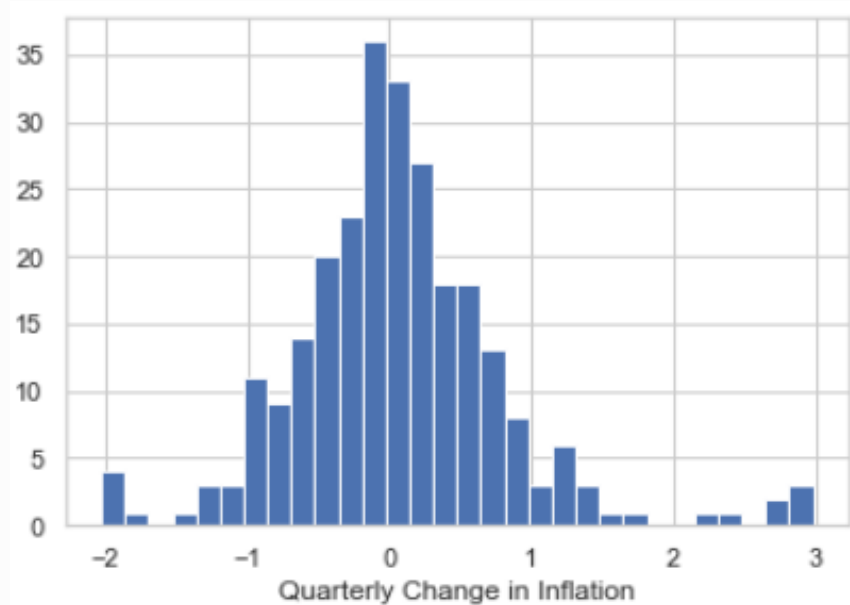


# Data Pre-Processing

Exploratory Data Analysis ( cont. )

## Investigating the Time Relationships ( cont. )

- We found the best of the 4 but **we** remembered that we **Winsorized Inflation on all; let's investigate what the variables showed without Winsorizing Inflation on our “best”**

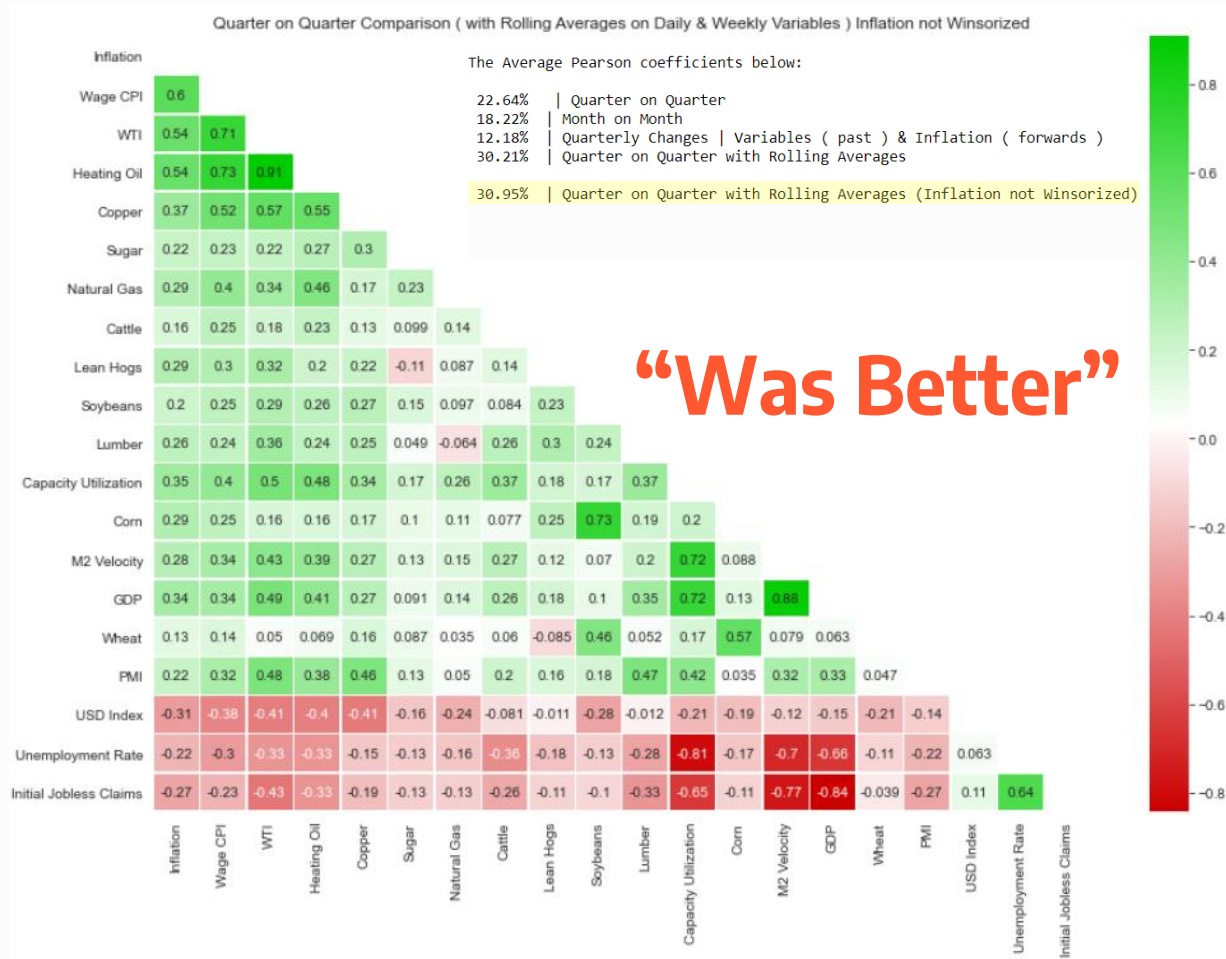




# Data Pre-Processing

Exploratory Data Analysis

Quarter on Quarter w/ Rolling Averages  
Feature Correlation Heat Maps with the  
Pearson correlation coefficients  
( cont. )

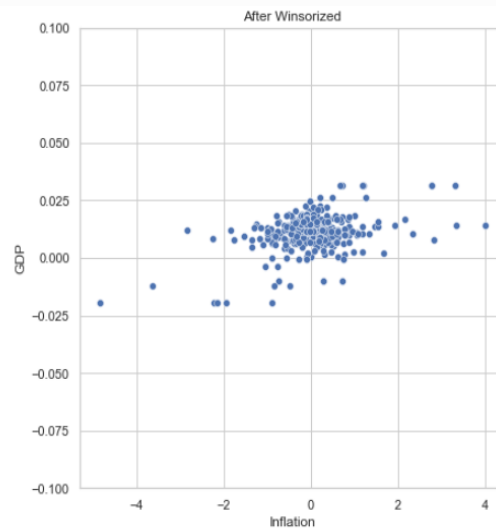
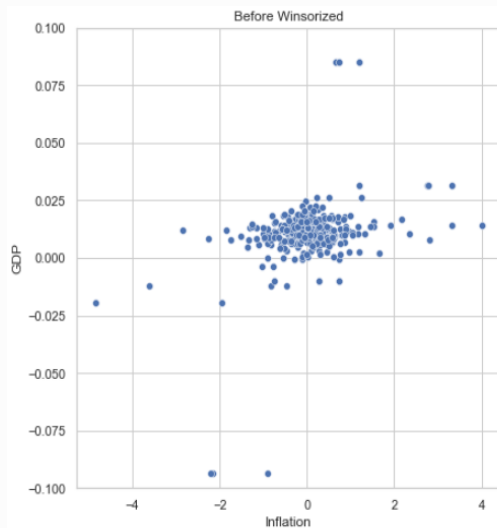


# Data Pre-Processing

Exploratory Data Analysis ( cont. )

## Investigating the Time Relationships ( cont. )

- We found the best of the 4 but we remembered that we Winsorized Inflation on all; let's investigate what the variables showed without Winsorizing Inflation
- Although **Winsorizing** did not work on Inflation, it **did work on 8 variables\***; this lead to an average increase in their Pearson correlation coefficients of 173 bps with one seeing a 460 bps increase



	Winsorized?
Wage CPI	n/a
WTI	Winsorized
Heating Oil	n/a
Copper	n/a
Sugar	Winsorized
Natural Gas	Winsorized
Cattle	n/a
Lean Hogs	n/a
Soybeans	n/a
Lumber	Winsorized
Capacity Utilization	Winsorized
Corn	n/a
M2 Velocity	n/a
GDP	Winsorized
Wheat	n/a
PMI	n/a
USD Index	n/a
Unemployment Rate	Winsorized
Initial Jobless Claims	Winsorized

\* Only one shown here; all are found in the Report

# Data Pre-Processing

Exploratory Data Analysis

Quarter on Quarter w/ Rolling Averages  
Feature Correlation Heat Maps with the  
Pearson correlation coefficients  
( cont. )



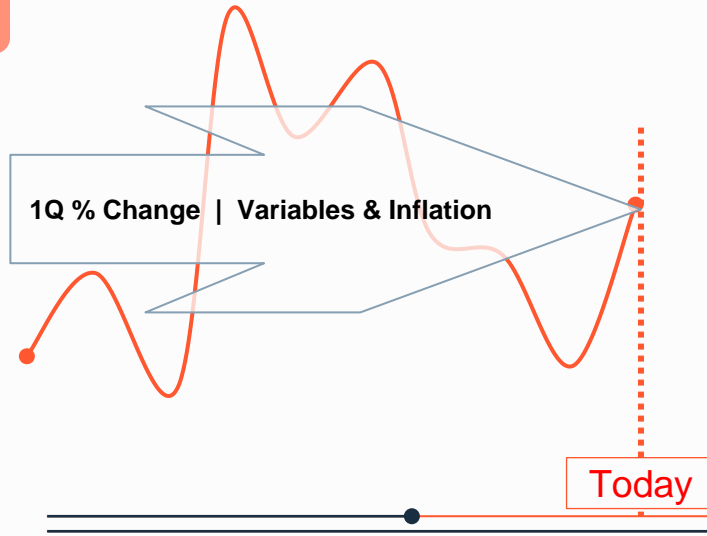
# Data Pre-Processing

Pre-Processing

## Splitting & Scaling

- **Chosen data frame**

- The Quarter on Quarter w/ Rolling Averages was chosen
  - Inflation not Winsorized but 8 are
- Train, Test Split
- Scaling

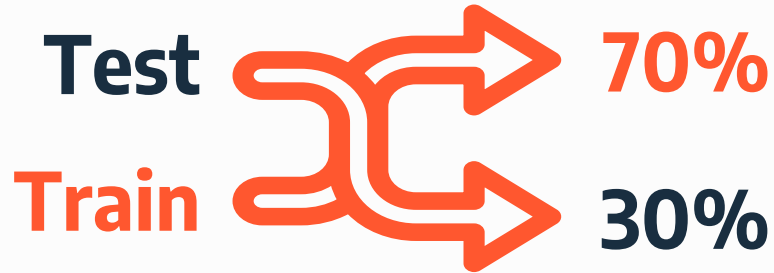


# Data Pre-Processing

Pre-Processing  
( cont. )

## Splitting & Scaling ( cont. )

- Chosen data frame
- **Train, Test Split**
  - The data was then split for Training & Testing to be sent to different Scaling Approaches
- Scaling

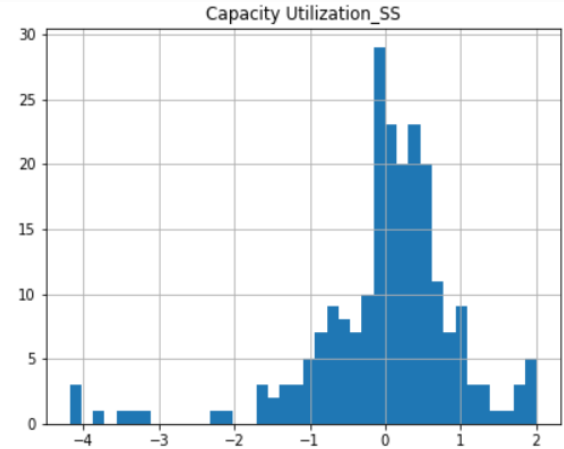
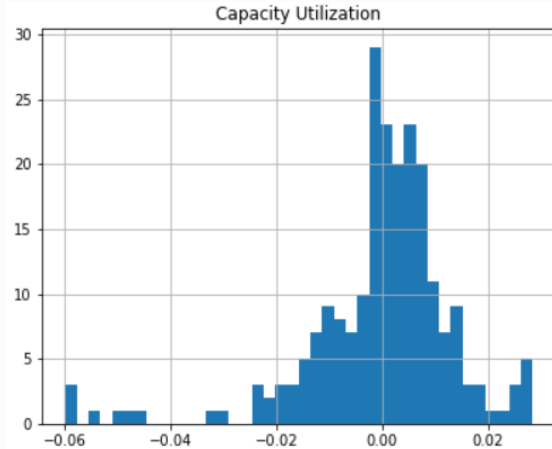


# Data Pre-Processing

Pre-Processing  
( cont. )

## Splitting & Scaling ( cont. )

- Chosen data frame
- Train, Test Split
- **Scaling**
  - 3 scaling approaches were tried to “normalize” the variables:
    - Standard Scaling ( SS )
    - MinMax Scaling ( MM )
    - Log Transformation ( LG )

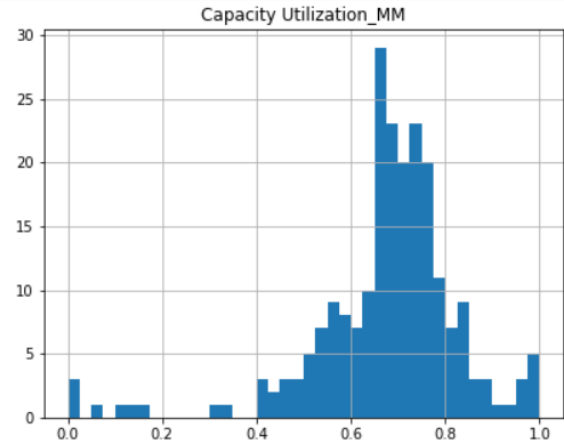
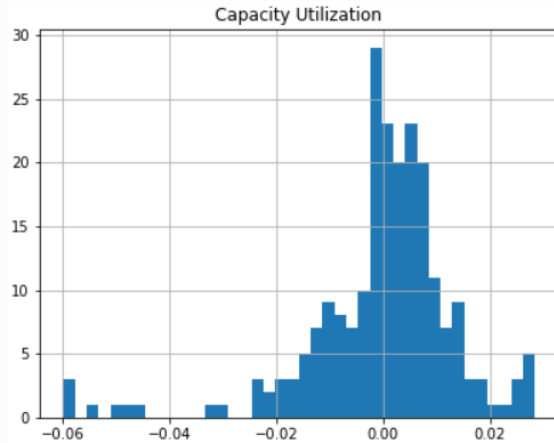


# Data Pre-Processing

Pre-Processing  
( cont. )

## Splitting & Scaling ( cont. )

- Chosen data frame
- Train, Test Split
- **Scaling**
  - 3 scaling approaches were tried to “normalize” the variables :
    - Standard Scaling ( SS )
    - MinMax Scaling ( MM )
    - Log Transformation ( LG )

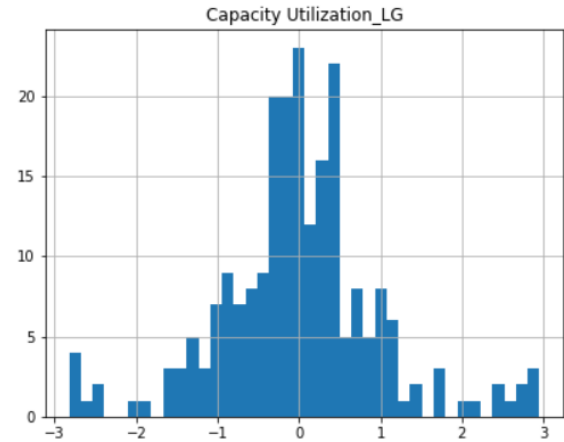
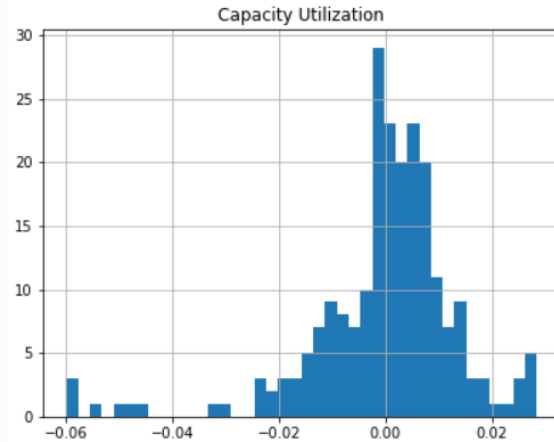


# Data Pre-Processing

Pre-Processing  
( cont. )

## Splitting & Scaling ( cont. )

- Chosen data frame
- Train, Test Split
- **Scaling**
  - 3 scaling approaches were tried to “normalize” the variables :
    - Standard Scaling ( SS )
    - MinMax Scaling ( MM )
  - Log Transformation ( LG )





# Data Pre-Processing

Pre-Processing  
( cont. )

## Data

## Splitting & Scaling ( cont. )

- Chosen data frame
- Train, Test Split
- **Scaling**
  - 3 scaling approaches were tried to “normalize” the variables :
    - Standard Scaling ( SS )
    - MinMax Scaling ( MM )
    - Log Transformation ( LG )

	Wages CPI_SS	WTI_SS	Wages CPI_MM	WTI_MM	Wages CPI_LG	WTI_LG
count	2.180000e+02	2.180000e+02	218.000000	218.000000	2.180000e+02	2.180000e+02
mean	-4.838128e-18	2.750094e-17	0.694134	0.579751	-2.340126e-16	-1.018553e-17
std	1.002301e+00	1.002301e+00	0.099718	0.153589	1.002301e+00	1.002301e+00
min	-6.977019e+00	-3.783391e+00	0.000000	0.000000	-4.203779e+00	-3.308051e+00
25%	-2.671202e-01	-5.665365e-01	0.667559	0.492937	-3.922100e-01	-6.014282e-01
50%	1.153214e-01	-3.959852e-02	0.705608	0.573683	2.665979e-02	-8.488108e-02
75%	4.280369e-01	6.677299e-01	0.736719	0.682071	3.947888e-01	6.501558e-01
max	3.074376e+00	2.742497e+00	1.000000	1.000000	4.675375e+00	3.071756e+00

# Data Pre-Processing

Pre-Processing  
( cont. )

## Splitting & Scaling ( cont. )

- Chosen data frame
- Train, Test Split
- **Scaling**
  - 3 scaling approaches were tried:
    - Standard Scaling ( SS )
    - MinMax Scaling ( MM )
    - Log Transformation ( LG )
  - MM posted poor results; thus removed

R<sup>2</sup> results for nothing scaled below  
Test 0.2925 ( nothing scaled )

R<sup>2</sup> results for X & y scaled below

SS Train	0.5055	Test 0.2962
MM Train	-6.3454	Test -6.8587
LG Train	0.4983	Test 0.2781

R<sup>2</sup> results for X only scaled below

SS Train	0.5133	Test 0.2925
MM Train	0.057	Test -0.042
LG Train	0.5005	Test 0.2732

R<sup>2</sup> results for the LG & SS combination below

SS Train	0.5053	Test 0.2788
----------	--------	-------------

R<sup>2</sup> averages of LG & SS X only scaled below

Av. Train	0.5069	Test 0.2828
-----------	--------	-------------

MAE results for nothing scaled below  
Test 0.5214 ( nothing scaled )

MAE results for X & y scaled below

SS Train	0.5085	Test 0.5859
MM Train	0.2581	Test 0.2538
LG Train	0.5172	Test 0.603

MAE results for X only scaled below

SS Train	0.4461	Test 0.5214
MM Train	0.5971	Test 0.6354
LG Train	0.4545	Test 0.5291

MAE results for the LG & SS combination below

SS Train	0.4488	Test 0.5229
----------	--------	-------------

MAE averages of LG & SS X only scaled below

Av. Train	0.4503	Test 0.5253
-----------	--------	-------------

MSE results for nothing scaled below  
Test 0.7133 ( nothing scaled )

MSE results for X & y scaled below

SS Train	0.4945	Test 0.6538
MM Train	0.0726	Test 0.0721
LG Train	0.5017	Test 0.6753

MSE results for X only scaled below

SS Train	0.3768	Test 0.5089
MM Train	0.7301	Test 0.7494
LG Train	0.3867	Test 0.5227

MSE results for the LG & SS combination below

SS Train	0.383	Test 0.5187
----------	-------	-------------

MSE averages of LG & SS X only scaled below

Av. Train	0.3818	Test 0.5158
-----------	--------	-------------

# Data Pre-Processing

Pre-Processing  
( cont. )

## Splitting & Scaling ( cont. )

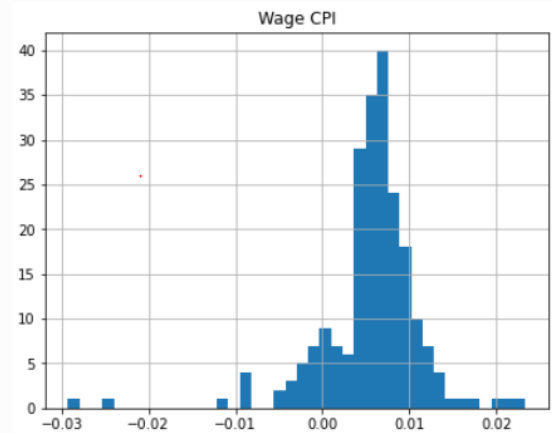
- Chosen data frame
- Train, Test Split
- **Scaling**
  - 3 scaling approaches were tried:
    - Standard Scaling ( SS )
    - ~~MinMax Scaling ( MM )~~
    - Log Transformation ( LG )
  - MM posted poor results; thus removed
  - As SS & LG posted the best result, variables were chosen to be sent to a new data frame for either a SS or LG while keeping the y variable ( Inflation ) unscaled.

# Data Pre-Processing

Pre-Processing  
( cont. )

## Splitting & Scaling ( cont. )

- Chosen data frame
- Train, Test Split
- **Scaling**
  - 3 scaling approaches were tried:
    - Standard Scaling ( SS )
    - MinMax Scaling ( MM )
    - Log Transformation ( LG )
  - MM posted poor results; thus removed
  - As SS & LG posted the best result, variables were chosen to be sent to a new data frame for either a SS or LG while keeping the y variable ( Inflation ) unscaled.
  - The following were sent to LG:
    - Wage CPI

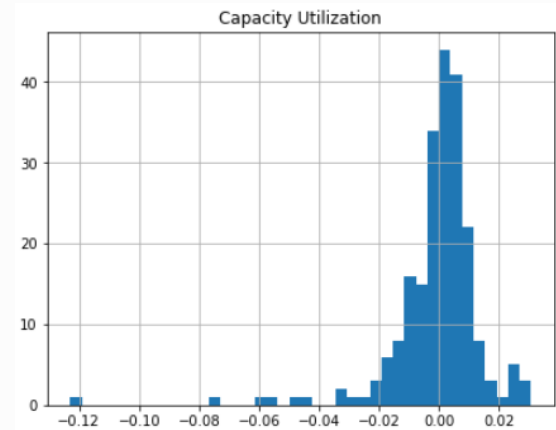


# Data Pre-Processing

Pre-Processing  
( cont. )

## Splitting & Scaling ( cont. )

- Chosen data frame
- Train, Test Split
- **Scaling**
  - 3 scaling approaches were tried:
    - Standard Scaling ( SS )
    - MinMax Scaling ( MM )
    - Log Transformation ( LG )
  - MM posted poor results; thus removed
  - As SS & LG posted the best result, variables were chosen to be sent to a new data frame for either a SS or LG while keeping the y variable ( Inflation ) unscaled.
  - The following were sent to LG:
    - Wage CPI
    - Capacity Utilization

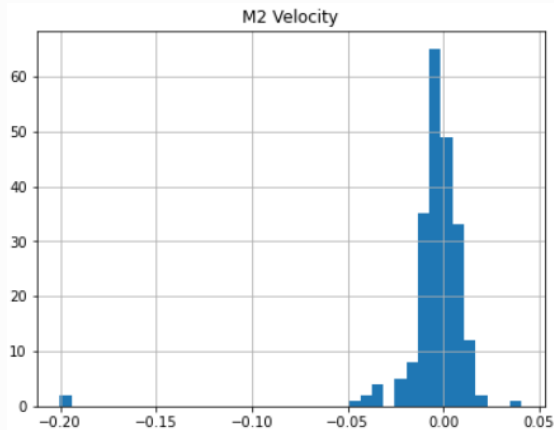


# Data Pre-Processing

Pre-Processing  
( cont. )

## Splitting & Scaling ( cont. )

- Chosen data frame
- Train, Test Split
- **Scaling**
  - 3 scaling approaches were tried:
    - Standard Scaling ( SS )
    - MinMax Scaling ( MM )
    - Log Transformation ( LG )
  - MM posted poor results; thus removed
  - As SS & LG posted the best result, variables were chosen to be sent to a new data frame for either a SS or LG while keeping the y variable ( Inflation ) unscaled.
  - The following were sent to LG:
    - Wage CPI
    - Capacity Utilization
    - M2 Velocity

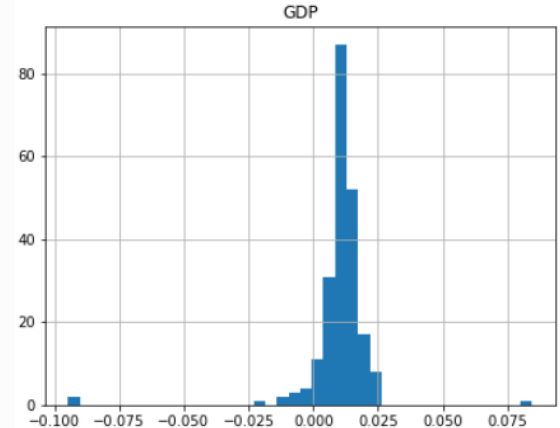


# Data Pre-Processing

Pre-Processing  
( cont. )

## Splitting & Scaling ( cont. )

- Chosen data frame
- Train, Test Split
- **Scaling**
  - 3 scaling approaches were tried:
    - Standard Scaling ( SS )
    - MinMax Scaling ( MM )
    - Log Transformation ( LG )
  - MM posted poor results; thus removed
  - As SS & LG posted the best result, variables were chosen to be sent to a new data frame for either a SS or LG while keeping the y variable ( Inflation ) unscaled.
  - The following were sent to LG:
    - Wage CPI
    - Capacity Utilization
    - M2 Velocity
    - GDP

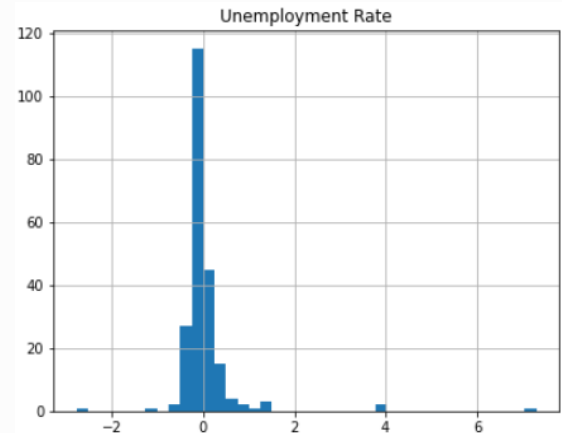


# Data Pre-Processing

Pre-Processing  
( cont. )

## Splitting & Scaling ( cont. )

- Chosen data frame
- Train, Test Split
- **Scaling**
  - 3 scaling approaches were tried:
    - Standard Scaling ( SS )
    - MinMax Scaling ( MM )
    - Log Transformation ( LG )
  - MM posted poor results; thus removed
  - As SS & LG posted the best result, variables were chosen to be sent to a new data frame for either a SS or LG while keeping the y variable ( Inflation ) unscaled.
  - The following were sent to LG:
    - Wage CPI
    - Capacity Utilization
    - M2 Velocity
    - GDP
    - Unemployment Rate





# Data Pre-Processing

Pre-Processing  
( cont. )

## Splitting & Scaling ( cont. )

- Chosen data frame
- Train, Test Split
- **Scaling**
  - 3 scaling approaches were tried:
    - Standard Scaling ( SS )
    - MinMax Scaling ( MM )
    - Log Transformation ( LG )
  - MM posted poor results; thus removed
  - As SS & LG posted the best result, variables were chosen to be sent to a new data frame for either a SS or LG while keeping the y variable ( Inflation ) unscaled.
  - The following were sent to LG:
    - Wage CPI
    - Capacity Utilization
    - M2 Velocity
    - GDP
    - Unemployment Rate
    - Initial Jobless Claims



# Data Pre-Processing

Pre-Processing  
( cont. )

## Splitting & Scaling ( cont. )

- Chosen data frame
- Train, Test Split
- **Scaling**
  - 3 scaling approaches were tried:
    - Standard Scaling ( SS )
    - MinMax Scaling ( MM )
    - Log Transformation ( LG )
  - MM posted poor results; thus removed
  - As SS & LG posted the best result, ""
  - The results of these below

R<sup>2</sup> results for nothing scaled below  
Test 0.2925 ( nothing scaled )

R<sup>2</sup> results for X & y scaled below

SS Train | 0.5055 Test 0.2962

~~MM Train | 6.3454 Test 6.0507~~

LG Train | 0.4983 Test 0.2781

MAE results for nothing scaled below  
Test 0.5214 ( nothing scaled )

MAE results for X & y scaled below

SS Train | 0.5085 Test 0.5859

~~MM Train | 0.2501 Test 0.2530~~

LG Train | 0.5172 Test 0.603

MSE results for nothing scaled below  
Test 0.7133 ( nothing scaled )

MSE results for X & y scaled below

SS Train | 0.4945 Test 0.6538

~~MM Train | 0.0726 Test 0.0721~~

LG Train | 0.5017 Test 0.6753

R<sup>2</sup> results for X only scaled below

SS Train | 0.5133 Test 0.2925

~~MM Train | 0.057 Test 0.042~~

LG Train | 0.5005 Test 0.2732

MAE results for X only scaled below

SS Train | 0.4461 Test 0.5214

~~MM Train | 0.3971 Test 0.0354~~

LG Train | 0.4545 Test 0.5291

MSE results for X only scaled below

SS Train | 0.3768 Test 0.5089

~~MM Train | 0.7301 Test 0.7494~~

LG Train | 0.3867 Test 0.5227

R<sup>2</sup> results for the LG & SS combination below

SS Train | 0.5053 Test 0.2788

MAE results for the LG & SS combination below

SS Train | 0.4488 Test 0.5229

MSE results for the LG & SS combination below

SS Train | 0.383 Test 0.5187

# Data Pre-Processing

Pre-Processing  
( cont. )

## Splitting & Scaling ( cont. )

- Chosen data frame
- Train, Test Split
- **Scaling**
  - 3 scaling approaches were tried:
    - Standard Scaling ( SS )
    - MinMax Scaling ( MM )
    - Log Transformation ( LG )
  - MM posted poor results; thus removed
  - As SS & LG posted the best result, ""
  - The results of these below
  - The resulting x5 Data frames went to a Random Forest Model

R<sup>2</sup> results for X & y scaled below

- 1 SS Train | 0.5055    Test 0.2962
- 2 LG Train | 0.4983    Test 0.2781

R<sup>2</sup> results for X only scaled below

- 3 SS Train | 0.5133    Test 0.2925
- 4 LG Train | 0.5005    Test 0.2732

R<sup>2</sup> results for the LG & SS combination below

- 5 SS Train | 0.5053    Test 0.2788

# 04



## Model Description

The Random Forest



# The **Random Forest Model** was then used

with the goal of determining what variables best explain & understand Inflation

# 05



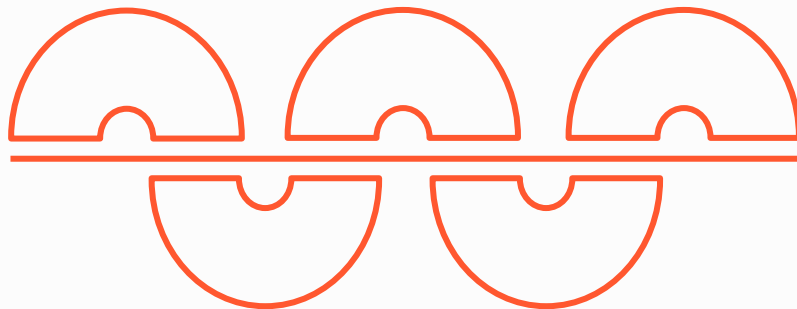
## Model Findings

What's moving Inflation?

# Model Findings

## Where's Inflation coming from?

- **The standard process was taken on x5**
  - Grid Search
  - Random Forest
  - Hyperparameter search using Grid Search CV

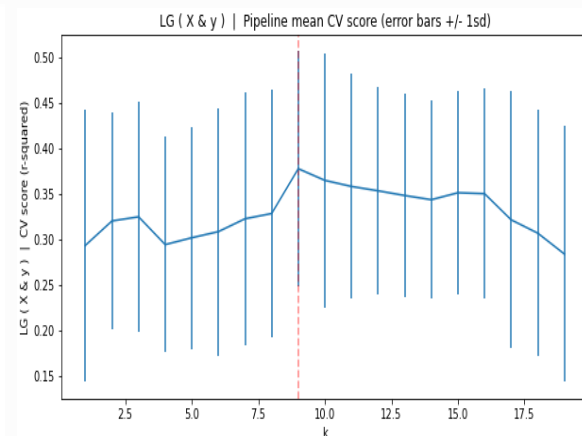
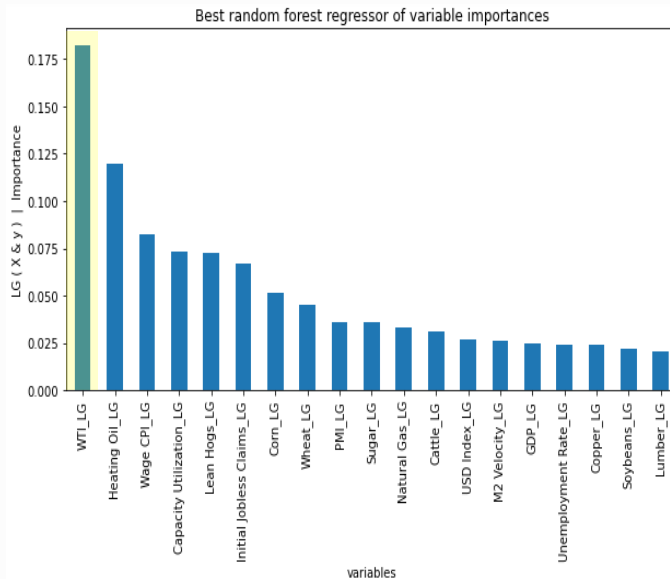


# Model Findings

( cont. )

## Where's Inflation coming from? ( cont. )

- The standard process was taken on x5
- **The results**
  - Random Forest showed **WTI holding a ubiquitous position as being the dominate Variable** on all scaling approaches



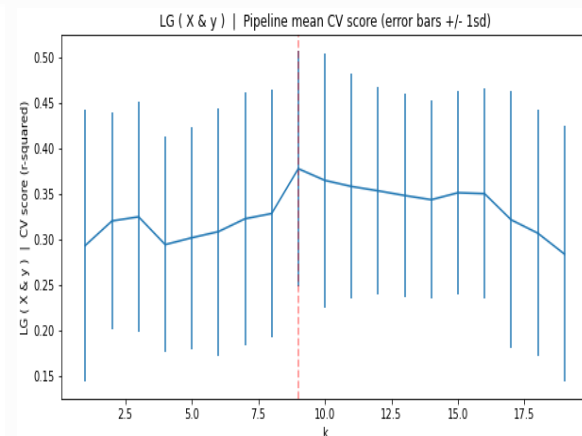
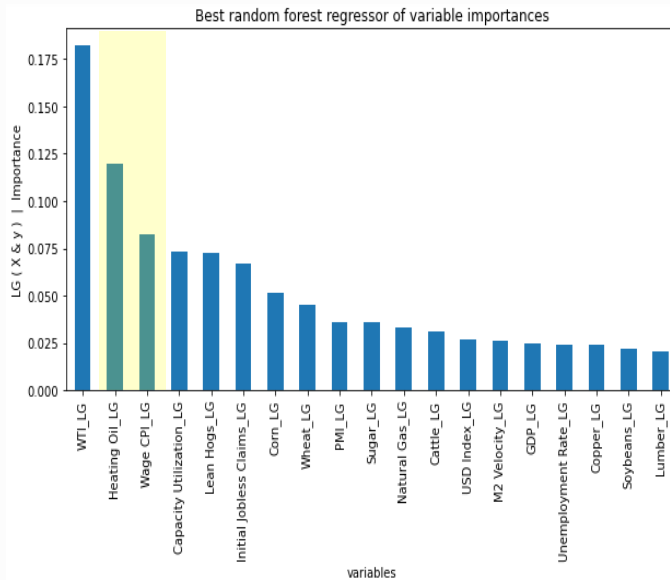


# Model Findings

( cont. )

## Where's Inflation coming from? ( cont. )

- The standard process was taken on x5
- **The results**
  - Random Forest showed WTI holding a ubiquitous position as being the dominate Variable on all scaling approaches; **Heating Oil & Wage CPI showed up in second & third place on many**

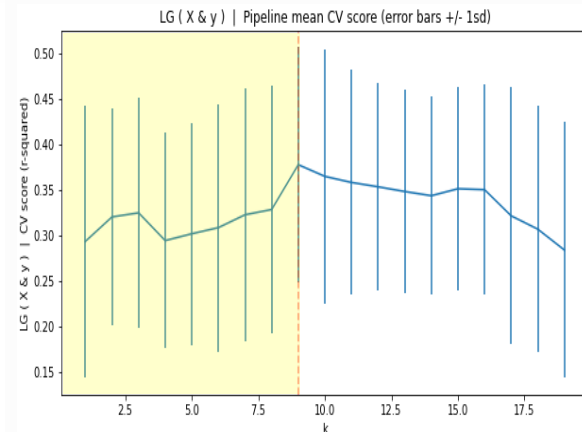
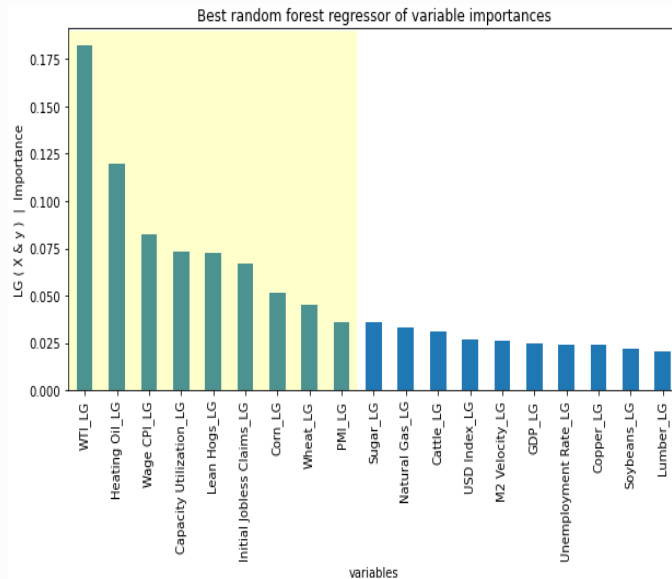


# Model Findings

( cont. )

## Where's Inflation coming from? ( cont. )

- The standard process was taken on x5
- **The results**
  - Random Forest showed WTI holding a ubiquitous position as being the dominate Variable on all scaling approaches; Heating Oil & Wage CPI showed up in second & third place on many. **Other variables helping varied\*; the below example has 9 variables**



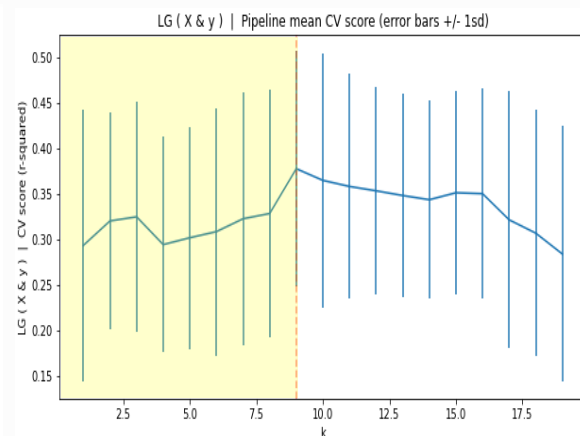
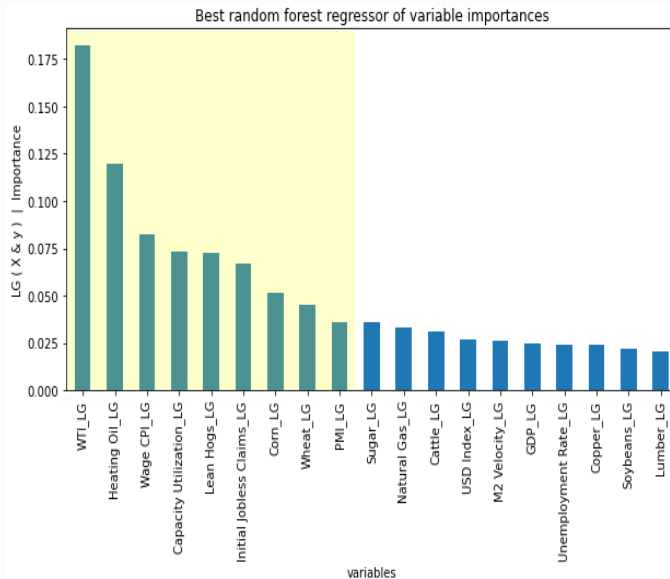
\* Only one shown here; all are found in the Report

# Model Findings

( cont. )

## Where's Inflation coming from? ( cont. )

- The standard process was taken on x5
- **The results**
  - Random Forest showed WTI holding a ubiquitous position as being the dominate Variable on all scaling approaches; Heating Oil & Wage CPI showed up in second & third place on many. Other variables helping varied\*; the below example has 9 variables
  - **It was then decided to isolate each to their respective variables**



# Model Findings

( cont. )

## Where's Inflation coming from? ( cont. )

- The standard process was taken on x5
- **The results**
  - Random Forest showed WTI holding a ubiquitous position as being the dominate Variable on all scaling approaches; Heating Oil & Wage CPI showed up in second & third place on many. Other variables helping varied\*; the below example has 9 variables
  - It was then decided to isolate each to their respective variables
  - In the end, the **LG approach on X only presented the best results**

R<sup>2</sup> results for X & y scaled below

SS Train | 0.492    Test 0.2706

LG Train | 0.4682    Test 0.2862

MAE results for X & y scaled below

SS Train | 0.5143    Test 0.6133

LG Train | 0.5261    Test 0.5955

MSE results for X & y scaled below

SS Train | 0.508    Test 0.6776

LG Train | 0.5318    Test 0.6676

R<sup>2</sup> results for X only scaled below

SS Train | 0.492    Test 0.2734

LG Train | 0.7563    Test 0.6524

MAE results for X only scaled below

SS Train | 0.4526    Test 0.6034

LG Train | 0.2229    Test 0.294

MSE results for X only scaled below

SS Train | 0.3933    Test 0.675

LG Train | 0.1886    Test 0.3251

R<sup>2</sup> results for the LG & SS combination below

SS Train | 0.4776    Test 0.2918

MAE results for the LG & SS combination below

SS Train | 0.2229    Test 0.294

MSE results for the LG & SS combination below

SS Train | 0.1886    Test 0.3251

# Model Findings

( cont. )

## Where's Inflation coming from? ( cont. )

- The standard process was taken on x5
- **The results**
  - Random Forest showed WTI holding a ubiquitous position as being the dominate Variable on all scaling approaches; Heating Oil & Wage CPI showed up in second & third place on many. Other variables helping varied\*; the below example has 9 variables
  - It was then decided to isolate each to their respective variables
  - In the end, the LG approach on X only presented the best results
  - & showed that the process presented notable improvement from Pre-processing

Comparing final to the averages in the Pre-processing Step

37.92 bps increase in  $R^2$

A -23.52 bps decrease in MAE

A -19.76 bps decrease in MSE

# Model Findings

( cont. )

## Where's Inflation coming from? ( cont. )

- The standard process was taken on x5
- **The results**
  - Random Forest showed WTI holding a ubiquitous position as being the dominate Variable on all scaling approaches; Heating Oil & Wage CPI showed up in second & third place on many. Other variables helping varied\*; the below example has 9 variables
  - It was then decided to isolate each to their respective variables
  - In the end, the LG approach on X only presented the best results
  - & showed that the process presented notable improvement from Pre-processing
  - **As WTI held the dominate place on all of the different structures of scaling to best position ourselves to understand Inflation; the verdict is...**

# Model Findings

( cont. )

## Where's Inflation coming from? ( cont. )

- The standard process was taken on x5
- **The results**
  - Random Forest showed WTI holding a ubiquitous position as being the dominate Variable on all scaling approaches; Heating Oil & Wage CPI showed up in second & third place on many. Other variables helping varied\*; the below example has 9 variables
  - It was then decided to isolate each to their respective variables
  - In the end, the LG approach on X only presented the best results
  - & showed that the process presented notable improvement from Pre-processing
  - As WTI held the dominate place on all of the different structures of scaling to best position ourselves to understand Inflation; the verdict is...
  - **We will borrow some words to help explain our verdict**

*The wise words of Bill Clintons' advisor to his 1992 political campaign*

“

”

*- James Carville*

*The wise words of Bill Clintons' advisor to his 1992 political campaign*


“It’s the economy, stupid”

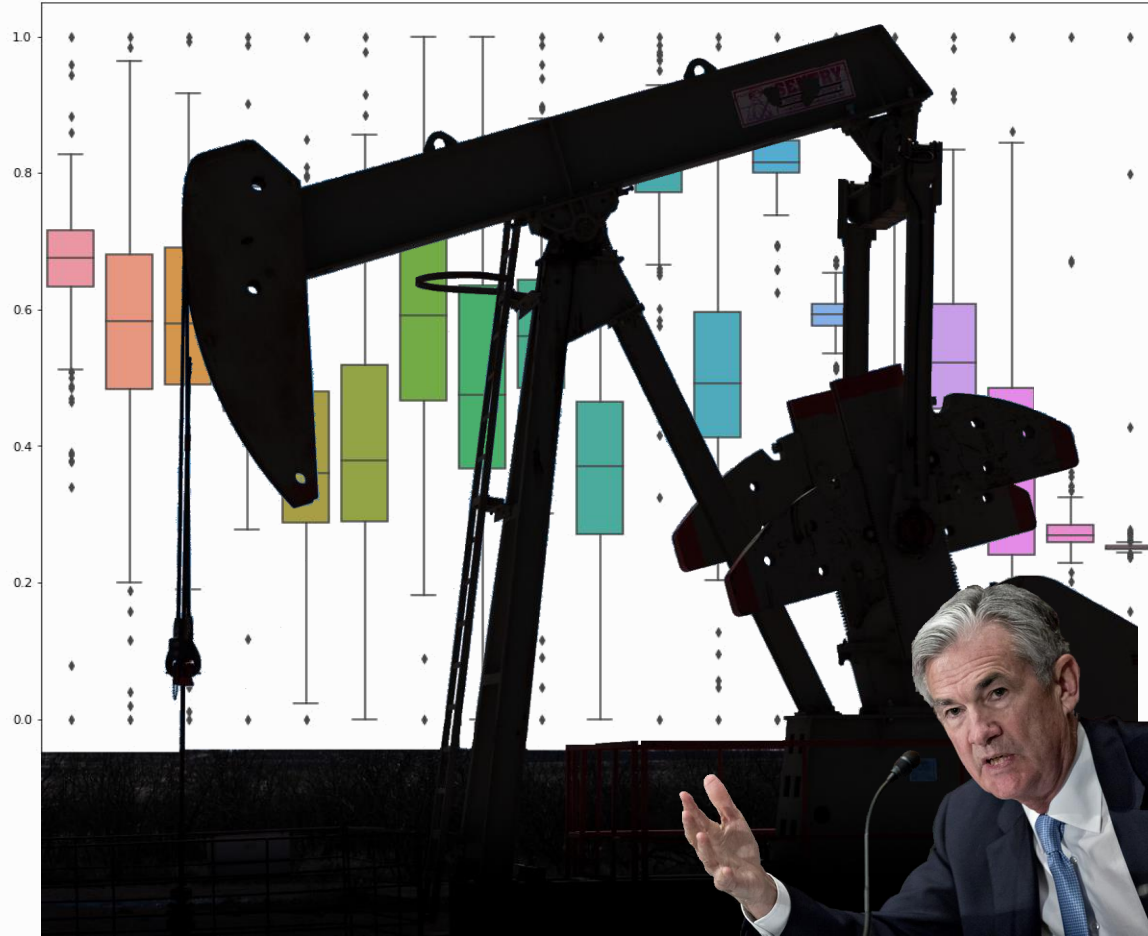
*- James Carville*



**Borrowed  
words...**



“It’s Oil, silly” 



**Our  
Conclusion**

# 06



## Next Steps

Keep going

## Next Steps

### Variables not included

- **Steel**
  - 2008 was the furthest I could pull

# Next Steps

## Variables not included

- Steel
- **Gasoline**
  - 2005 was the furthest I could pull

# Next Steps

## Variables not included

- Steel
- Gasoline
- **US Wages Hourly Earnings**
  - Limited Data as well

# Next Steps

## Variables not included

- Steel
- Gasoline
- US Wages Hourly Earnings
- **US Dollar Index: Broad, Goods & Services**
  - Only goes until 2006

# Next Steps

## Variables not included

- Steel
- Gasoline
- US Wages Hourly Earnings
- US Dollar Index: Broad, Goods & Services
- **Growth in M2**
  - Possible overlap with M2 Velocity

# Next Steps

( cont. )

**More attention may be applicable to the below:**

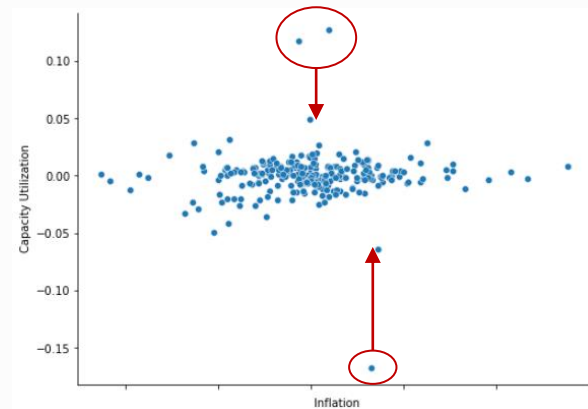
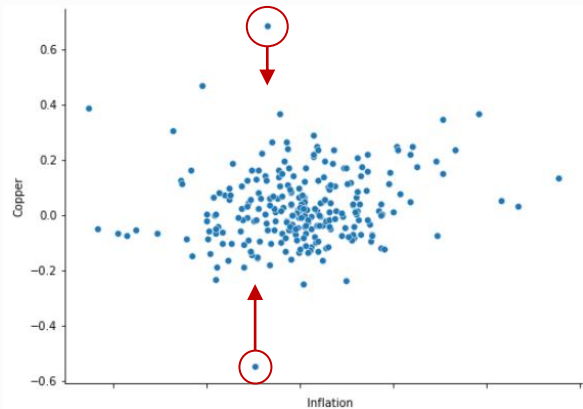
- **Get more data**
  - The big set back would be the size of the data frame. With only 321 observations, machine learning is limited



# Next Steps (cont.)

## More attention may be applicable to the below:

- Get more data
- **Winsorizing**
  - Winsorization on Inflation & other variables may be re-examined



# Next Steps

( cont. )

## More attention may be applicable to the below:

- Get more data
- Winsorizing
- **The SS & LG Divide**
  - Reassess the Variables which were chosen in the SS & LG divide; discussed in Pre-processing

# Next Steps

( cont. )

## More attention may be applicable to the below:

- Get more data
- Winsorizing
- The SS & LG Divide
- **Predict Wages CPI Itself**
  - Develop a model to remove ourselves from the US govt's reporting

# Next Steps

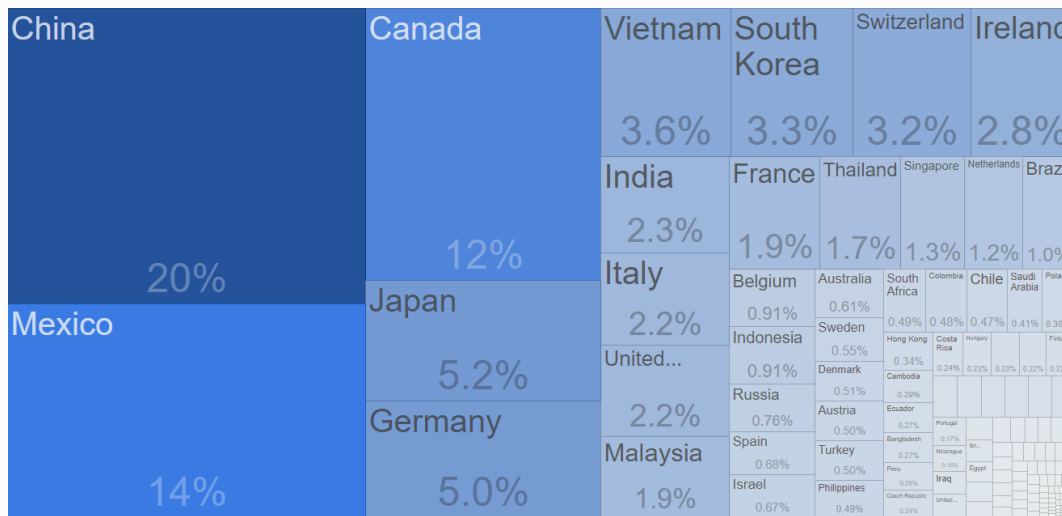
( cont. )

## More attention may be applicable to the below:

- Get more data
- Winsorizing
- The SS & LG Divide
- Predict Wages CPI Itself
- **Build a Better Imported / Exported USD**
  - **The DXY doesn't correctly address the US Imports or Exports Inflation** as it's weighting is a weighted geometric mean of the:
    - Eurozone ( EUR ),
    - Japan ( JPY ),
    - United Kingdom ( GBP ),
    - Canada ( CAD ),
    - Sweden ( SEK ) &
    - Switzerland ( CHF )

- Get more data
- Winsorizing
- The SS & LG Divide
- Predict Wages CPI Itself

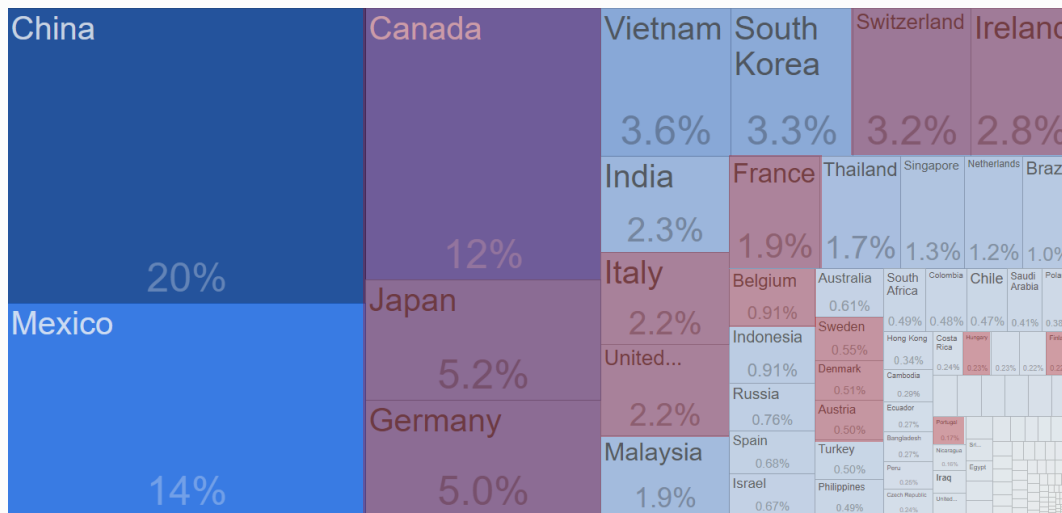
- The DXY doesn't correctly address the US Imports or Exports Inflation as it's weighting is a weighted geometric mean of various currencies
- **Doesn't take into account the US's largest trading partner, China. Imports shown below**



# Next Steps (cont.)

## More attention may be applicable to the below:

- Get more data
- Winsorizing
- The SS & LG Divide
- Predict Wages CPI Itself
- **Build a Better Imported / Exported USD**
  - The DXY doesn't correctly address the US Imports or Exports Inflation as it's weighting is a weighted geometric mean of various currencies
  - Doesn't take into account the US's largest trading partner, China. Imports shown below.
  - **It takes into account less than 40% of US Import Trade**



# Next Steps

( cont. )

## More attention may be applicable to the below:

- Get more data
- Winsorizing
- The SS & LG Divide
- Predict Wages CPI Itself
- Build a Better Imported / Exported USD
- **Random Forest was used, while Gradient Boosting may be something to explore:**
  - i.e. Boosting over Bagging

# Thanks

By **Rand Sobczak Jr.**  
rand.sobczak@gmail.com  
+1 313 447 8634



CREDITS: This presentation template was created by **Slidesgo**, including icons by **Flaticon**, and infographics & images by **Freepik**





Questions?