Consulting Project

- Hurricane -

Consulting Project - Hurricane -

On August 31, 2016 Hurricane Hermine struck the East cost. More than $8 billion in federal disaster relief and insurance money came into the county, resulting in increased sales at department stores and numerous other businesses.

The Carlson Department Store suffered heavy damage. It was closed for four months (September through December). Carlson is now involved in a dispute with its insurance company about the amount of lost sales during the time the store was closed.

Two key issues must be resolved

1. The amount of sales Carlson had made if the hurricane had not struck.
2. Whether Carlson is entitled to an additional compensation for the loss of additional ("excess") sales due to increased business activity after the storm.

Consulting Project - Hurricane -

.

TABLE 1. COUNTY SALES (millions)

	January	February	March	April	May	June	July	August	September	October	November	December
2016									55.8	56.4	71.4	117.6
2017	46.8	48	60	57.6	61.8	58.2	56.4	63	57.6	53.4	71.4	114
2018	46.8	48.6	59.4	58.2	60.6	55.2	51	58.8	49.8	54.6	65.4	102
2019	43.8	45.6	57.6	53.4	56.4	52.8	54	60.6	47.4	54.6	67.8	100.2
2020	48	51.6	57.6	58.2	60	57	57.6	61.8	69	75	85.2	121.8

TABLE 2. STORE SALES (millions)

	January	February	March	April	May	June	July	August	September	October	November	December
2016									1.71	1.9	2.74	4.2
2017	1.45	1.8	2.03	1.99	2.32	2.2	2.13	2.43	1.9	2.13	2.56	4.16
2018	2.31	1.89	2.02	2.23	2.39	2.14	2.27	2.21	1.89	2.29	2.83	4.04
2019	2.31	1.99	2.42	2.45	2.57	2.42	2.4	2.5	2.09	2.54	2.97	4.35
2020	2.56	2.28	2.69	2.48	2.73	2.37	2.31	2.23		lost	sales	

Consulting Project - Hurricane -

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import statsmodels.formula.api as smf
```

```
df0 = pd.read_csv('hurricane.csv')
```

```
df0[:12]
```

	Month	Year	Carlson	County
2016-09-01	September	2016	1.71	55.8
2016-10-01	October	2016	1.90	56.4
2016-11-01	November	2016	2.74	71.4
2016-12-01	December	2016	4.20	117.6
2017-01-01	January	2017	1.45	46.8
2017-02-01	February	2017	1.80	48.0
2017-03-01	March	2017	2.03	60.0
2017-04-01	April	2017	1.99	57.6
2017-05-01	May	2017	2.32	61.8
2017-06-01	June	2017	2.20	58.2
2017-07-01	July	2017	2.13	56.4
2017-08-01	August	2017	2.43	63.0

Consulting Project - Hurricane

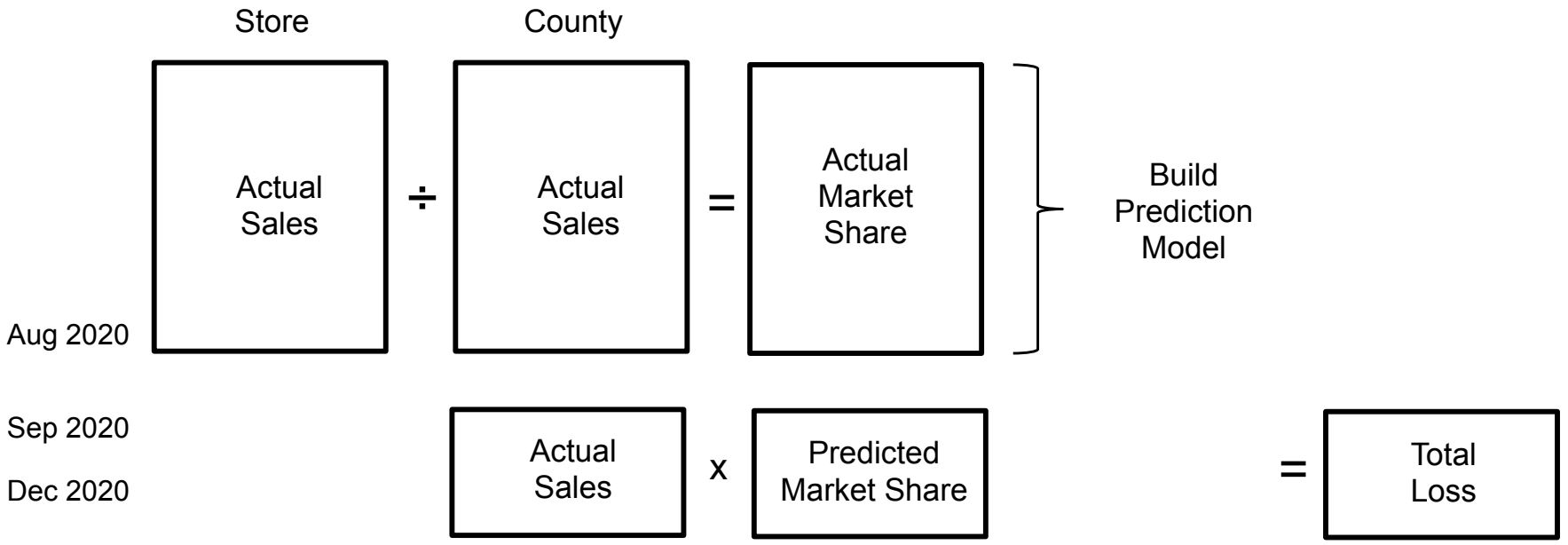
Approach 1

Approach 1

- Find Carlson store market share (the fraction of Carlson monthly sales to the county-wide department stores monthly sales)
- Build a regression model to predict Carlson's market share from Sep 2020 to Dec 2020
- Multiply Carlson's predicted market share by the actual county department store sales month by month
- These are the monthly total sales (regular and hurricane induced) lost by Carlson Store

Approach 1

.


Store
County
Actual Sales
÷
Actual Sales
=
Actual Market Share
Build Prediction Model
Aug 2020
Sep 2020
Dec 2020
Actual Sales
x
Predicted Market Share
=
Total Loss

Approach 1

```
df = df0.copy()
```

```
df.tail(9)
```

	Month	Year	Carlson	County
2020-04-01	April	2020	2.48	58.2
2020-05-01	May	2020	2.73	60.0
2020-06-01	June	2020	2.37	57.0
2020-07-01	July	2020	2.31	57.6
2020-08-01	August	2020	2.23	61.8
2020-09-01	September	2020	NaN	69.0
2020-10-01	October	2020	NaN	75.0
2020-11-01	November	2020	NaN	85.2
2020-12-01	December	2020	NaN	121.8

APPROACH 1 – Find Actual Market Share values

```
df['Carl_Mkt_share'] = df['Carlson']/df['County']
df.tail(9)
```

	Month	Year	Carlson	County	Carl_Mkt_share
2020-04-01	April	2020	2.48	58.2	0.042612
2020-05-01	May	2020	2.73	60.0	0.045500
2020-06-01	June	2020	2.37	57.0	0.041579
2020-07-01	July	2020	2.31	57.6	0.040104
2020-08-01	August	2020	2.23	61.8	0.036084
2020-09-01	September	2020	NaN	69.0	NaN
2020-10-01	October	2020	NaN	75.0	NaN
2020-11-01	November	2020	NaN	85.2	NaN
2020-12-01	December	2020	NaN	121.8	NaN

APPROACH 1 – Predict Market Share

```
model_Mkt_share = smf.ols('Carl_Mkt_share ~ Year + C(Month)',
                          data = df).fit()
```

	Month	Year	Carl_Mkt_share
2020-04-01	April	2020	0.042612
2020-05-01	May	2020	0.045500
2020-06-01	June	2020	0.041579
2020-07-01	July	2020	0.040104
2020-08-01	August	2020	0.036084
2020-09-01	September	2020	NaN
2020-10-01	October	2020	NaN
2020-11-01	November	2020	NaN
2020-12-01	December	2020	NaN

APPROACH 1 – Predict Market Share

```
model_Mkt_share = smf.ols('Carl_Mkt_share ~ Year + C(Month)',
                          data = df).fit()

df['Carl_Mkt_share_pred'] = model_Mkt_share.predict(df)
df.tail(9)
```

	Month	Year	Carl_Mkt_share	Carl_Mkt_share_pred
2020-04-01	April	2020	0.042612	0.044851
2020-05-01	May	2020	0.045500	0.046524
2020-06-01	June	2020	0.041579	0.045507
2020-07-01	July	2020	0.040104	0.046218
2020-08-01	August	2020	0.036084	0.042886
2020-09-01	September	2020	NaN	0.043939
2020-10-01	October	2020	NaN	0.048029
2020-11-01	November	2020	NaN	0.047847
2020-12-01	December	2020	NaN	0.046326

APPROACH 1 – Predict Market Share

```
model_Mkt_share = smf.ols('Carl_Mkt_share ~ Year + C(Month)',
                          data = df).fit()

df['Carl_Mkt_share_pred'] = model_Mkt_share.predict(df)
df.tail(9)
```

	Month	Year	Carlson	County	Carl_Mkt_share	Carl_Mkt_share_pred
2020-04-01	April	2020	2.48	58.2	0.042612	0.044851
2020-05-01	May	2020	2.73	60.0	0.045500	0.046524
2020-06-01	June	2020	2.37	57.0	0.041579	0.045507
2020-07-01	July	2020	2.31	57.6	0.040104	0.046218
2020-08-01	August	2020	2.23	61.8	0.036084	0.042886
2020-09-01	September	2020	NaN	69.0	NaN	0.043939
2020-10-01	October	2020	NaN	75.0	NaN	0.048029
2020-11-01	November	2020	NaN	85.2	NaN	0.047847
2020-12-01	December	2020	NaN	121.8	NaN	0.046326

APPROACH 1 – Predict Market Share

```
begin = "2020-09-01"
df2 = df.loc[begin:,].copy()
df2
```

	Month	Year	Carlson	County	Carl_Mkt_share	Carl_Mkt_share_pred
2020-09-01	September	2020	NaN	69.0	NaN	0.043939
2020-10-01	October	2020	NaN	75.0	NaN	0.048029
2020-11-01	November	2020	NaN	85.2	NaN	0.047847
2020-12-01	December	2020	NaN	121.8	NaN	0.046326

APPROACH 1 – Predict Market Share

```
df2.drop(['Carlson','Carl_Mkt_share'],axis=1,inplace=True)
df2
```

	Month	Year	County	Carl_Mkt_share_pred
2020-09-01	September	2020	69.0	0.043939
2020-10-01	October	2020	75.0	0.048029
2020-11-01	November	2020	85.2	0.047847
2020-12-01	December	2020	121.8	0.046326

```
df2['loss'] = df2['County'] * df2['Carl_Mkt_share_pred']
df2
```

	Month	Year	County	Carl_Mkt_share_pred		loss
2020-09-01	September	2020	69.0	0.043939		3.031775
2020-10-01	October	2020	75.0	0.048029	=	3.602180
2020-11-01	November	2020	85.2	0.047847		4.076529
2020-12-01	December	2020	121.8	0.046326		5.642558

X

APPROACH 1 – Predict Market Share

```
df2.drop(['Carlson','Carl_Mkt_share'],axis=1,inplace=True)
df2
```

	Month	Year	County	Carl_Mkt_share_pred
2020-09-01	September	2020	69.0	0.043939
2020-10-01	October	2020	75.0	0.048029
2020-11-01	November	2020	85.2	0.047847
2020-12-01	December	2020	121.8	0.046326

```
df2['loss'] = df2['County'] * df2['Carl_Mkt_share_pred']
df2
```

	Month	Year	County	Carl_Mkt_share_pred		loss
2020-09-01	September	2020	69.0	0.043939		3.031775
2020-10-01	October	2020	75.0	0.048029	=	3.602180
2020-11-01	November	2020	85.2	0.047847		4.076529
2020-12-01	December	2020	121.8	0.046326		5.642558
				`df2.loss.sum()`		16.353041

X

Carlson Dept. Stores can claim a loss of $ 16,353,041

APPROACH 1 – Predict Market Share

```
model_Mkt_share = smf.ols('Carl_Mkt_share ~ Year + C(Month)',
                          data = df).fit()
model_Mkt_share.summary()
```

OLS Regression Results

Dep. Variable:	Carl_Mkt_share	**R-squared:**	0.654
Model:	OLS	**Adj. R-squared:**	0.536
Method:	Least Squares	**F-statistic:**	5.517
		Prob (F-statistic):	3.53e-05
Time:	18:01:34	**Log-Likelihood:**	210.85
No. Observations:	48	**AIC:**	-395.7
Df Residuals:	35	**BIC:**	-371.4

Consulting Project - Hurricane

Approach 2

APPROACH 2

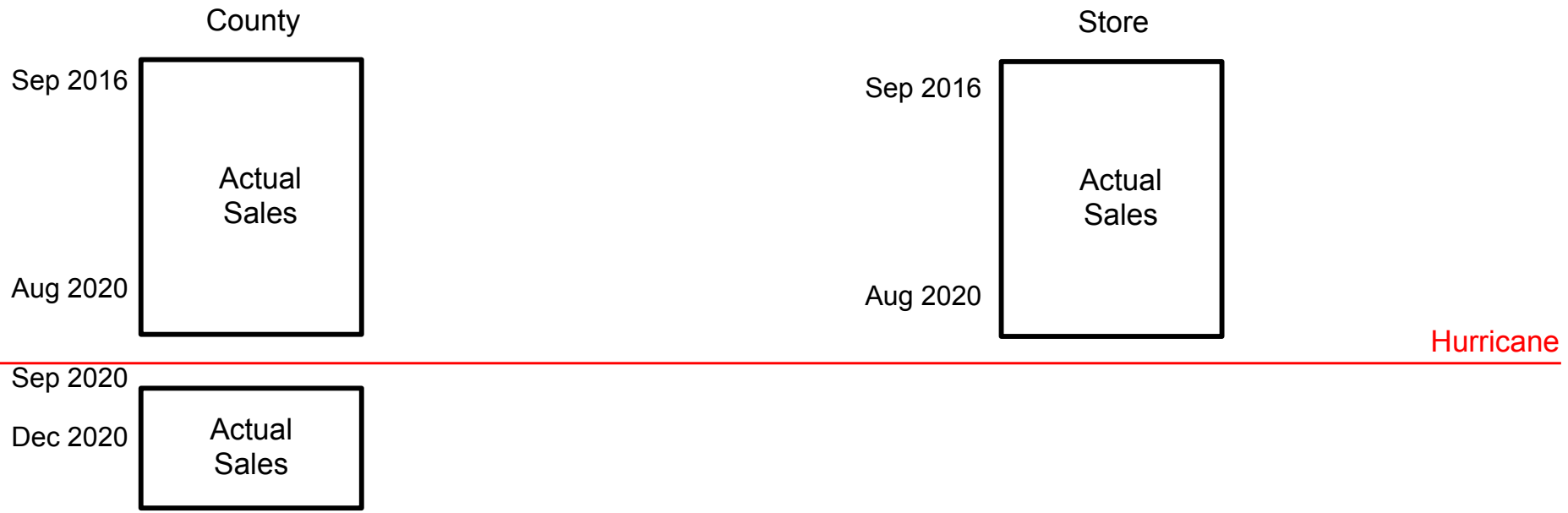
- Build a regression model to predict County Sales
- Compare County Sales (actual vs. predicted)
- Find County stores Excess Sales ratios (actual ÷ predicted)

APPROACH 2

- Build a regression model to predict County Sales
- Compare County Sales (actual vs. predicted)
- Find County stores Excess Sales ratios (actual ÷ predicted)
- Build a regression model to predict Carlson's store regular sales
- Multiply County stores Excess Sales ratios by the predicted Carlson's store regular sales
- The result are the monthly Total sales (regular and induced) lost by Carlson's store

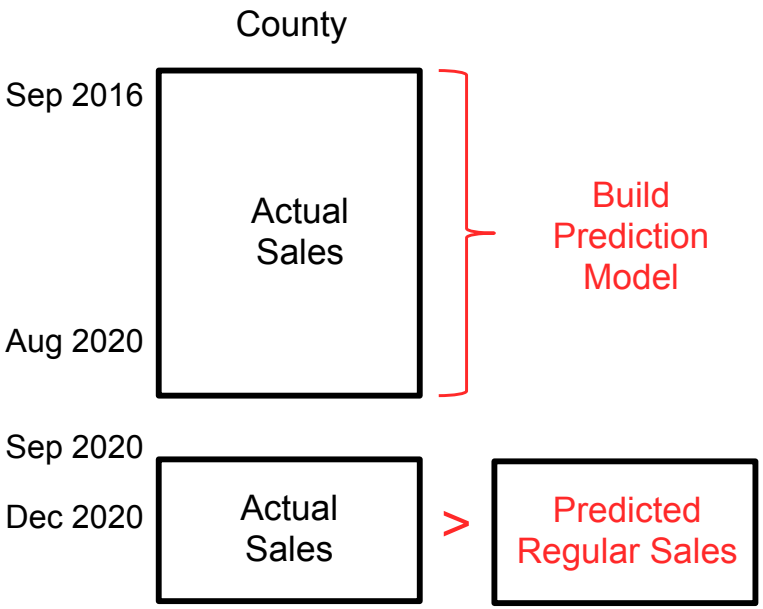
APPROACH 2

.


County
Sep 2016
Actual Sales
Aug 2020
Store
Sep 2016
Actual Sales
Aug 2020
Hurricane
Sep 2020
Dec 2020
Actual Sales

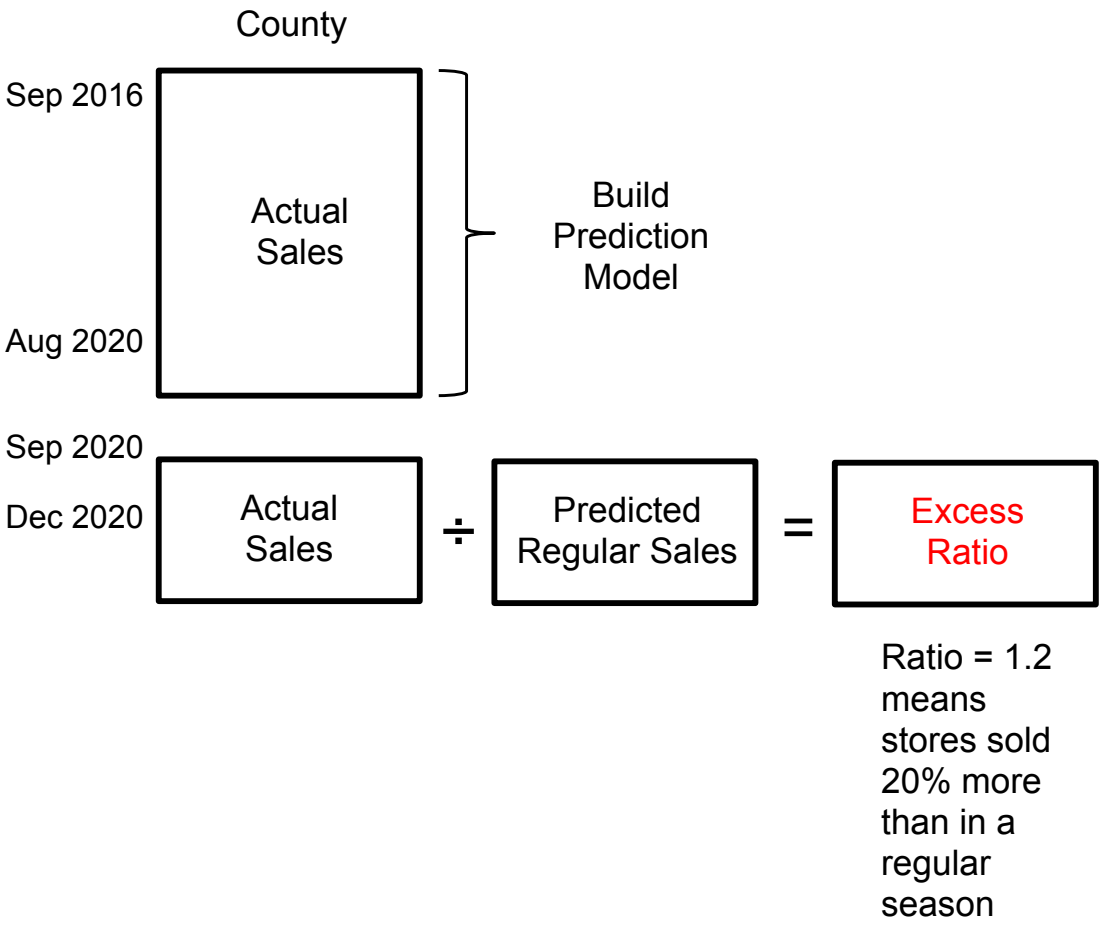
APPROACH 2

.


County
Sep 2016
Actual Sales
Aug 2020
Build Prediction Model
Sep 2020
Dec 2020
Actual Sales
>
Predicted Regular Sales

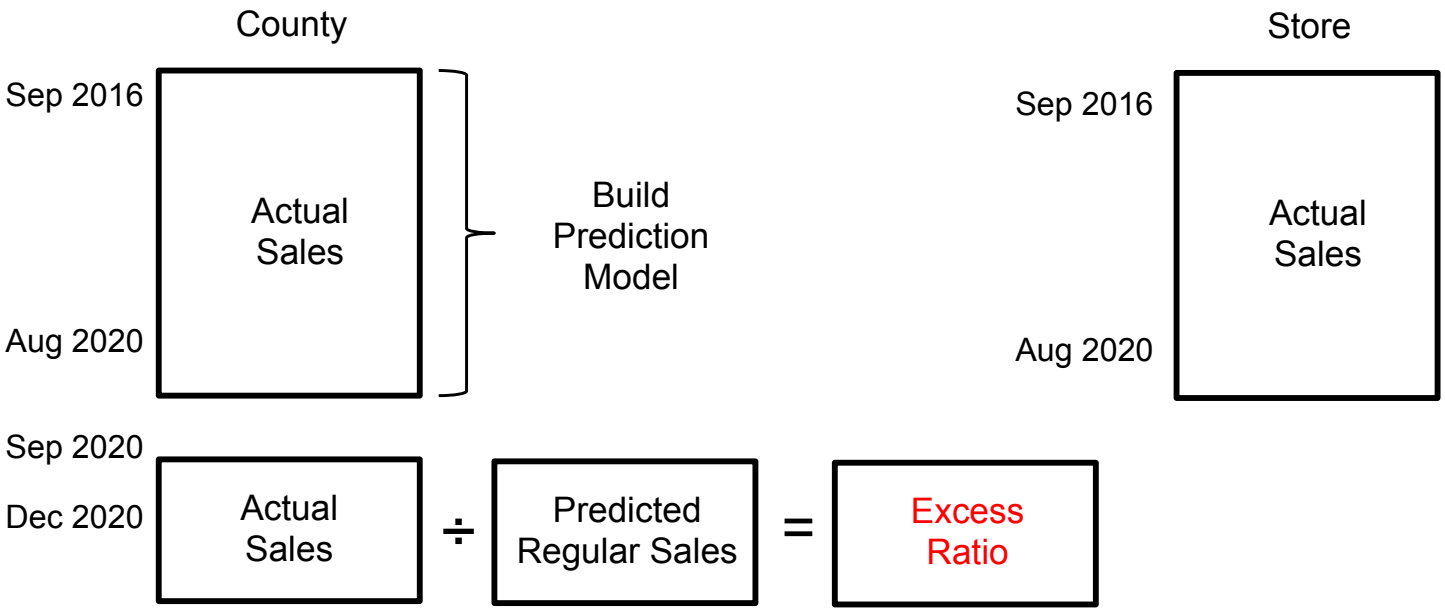
APPROACH 2

.


County
Sep 2016
Aug 2020
Actual Sales
Build Prediction Model
Sep 2020
Dec 2020
Actual Sales
÷
Predicted Regular Sales
=
Excess Ratio
Ratio = 1.2 means stores sold 20% more than in a regular season

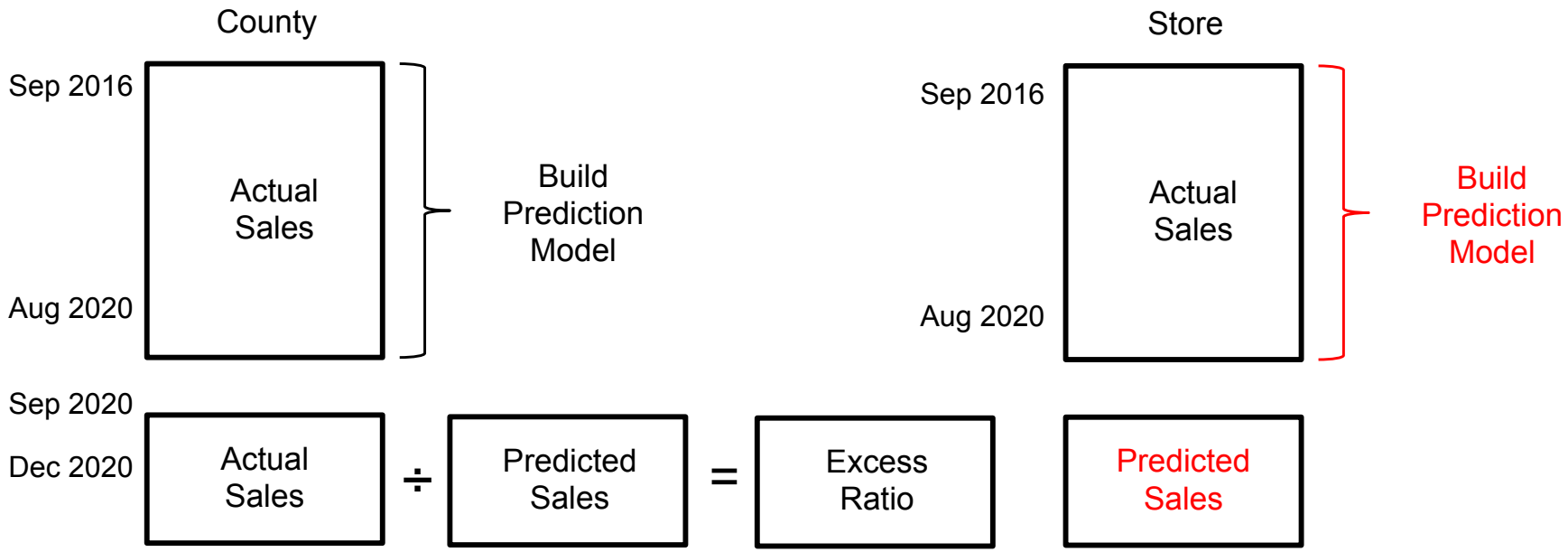
APPROACH 2

.


County
Sep 2016
Actual Sales
Aug 2020
Build Prediction Model
Store
Sep 2016
Actual Sales
Aug 2020
Sep 2020
Dec 2020
Actual Sales
÷
Predicted Regular Sales
=
Excess Ratio

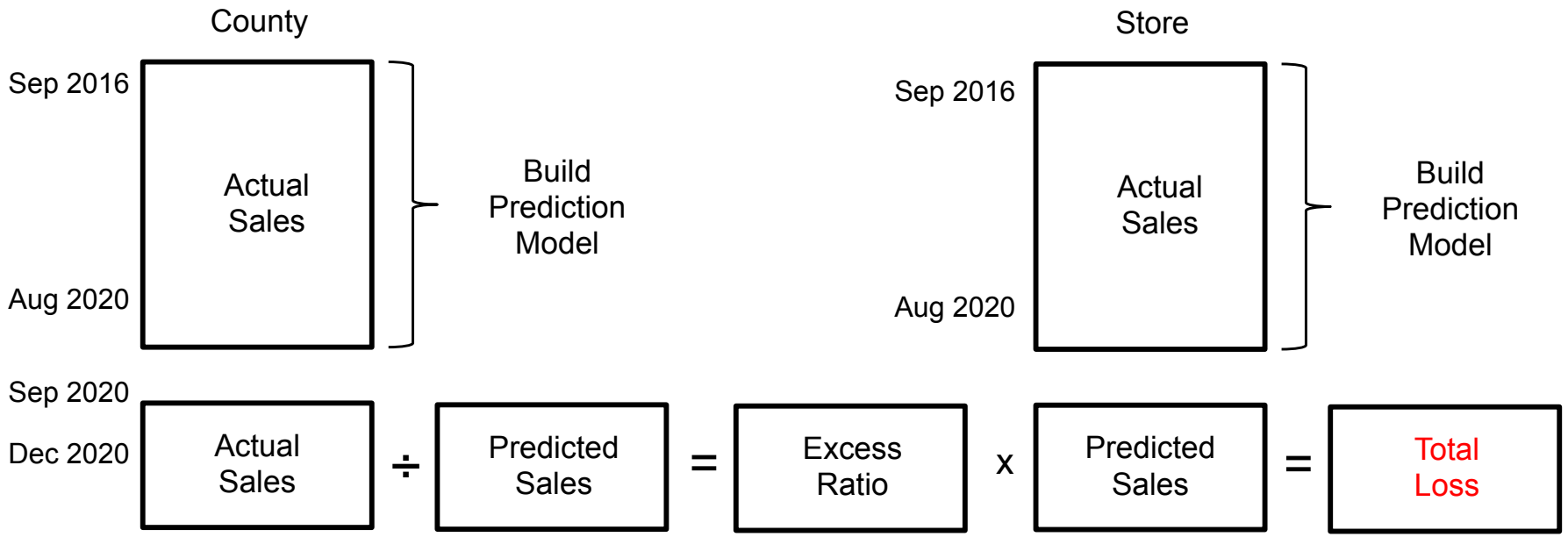
EXAMPLE 2 – APPROACH 2

.



APPROACH 2

.


County
Sep 2016
Aug 2020
Actual Sales
Build Prediction Model
Store
Sep 2016
Aug 2020
Actual Sales
Build Prediction Model
Sep 2020
Dec 2020
Actual Sales
÷
Predicted Sales
=
Excess Ratio
x
Predicted Sales
=
Total Loss

APPROACH 2

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import statsmodels.formula.api as smf
```

```
df0 = pd.read_csv('hurricane.csv')
```

```
df[:12]
```

	Month	Year	Carlson	County
2016-09-01	September	2016	1.71	55.8
2016-10-01	October	2016	1.90	56.4
2016-11-01	November	2016	2.74	71.4
2016-12-01	December	2016	4.20	117.6
2017-01-01	January	2017	1.45	46.8
2017-02-01	February	2017	1.80	48.0
2017-03-01	March	2017	2.03	60.0
2017-04-01	April	2017	1.99	57.6
2017-05-01	May	2017	2.32	61.8
2017-06-01	June	2017	2.20	58.2
2017-07-01	July	2017	2.13	56.4
2017-08-01	August	2017	2.43	63.0

APPROACH 2 – Split data set into two dataframes

```
Carlson = df.drop(["County"],axis=1)
Carlson.tail(9)
```

	Month	Year	Carlson
2020-04-01	April	2020	2.48
2020-05-01	May	2020	2.73
2020-06-01	June	2020	2.37
2020-07-01	July	2020	2.31
2020-08-01	August	2020	2.23
2020-09-01	September	2020	NaN
2020-10-01	October	2020	NaN
2020-11-01	November	2020	NaN
2020-12-01	December	2020	NaN

```
County = df.drop(["Carlson"],axis=1)
County.tail(9)
```

	Month	Year	County
2020-04-01	April	2020	58.2
2020-05-01	May	2020	60.0
2020-06-01	June	2020	57.0
2020-07-01	July	2020	57.6
2020-08-01	August	2020	61.8
2020-09-01	September	2020	69.0
2020-10-01	October	2020	75.0
2020-11-01	November	2020	85.2
2020-12-01	December	2020	121.8

APPROACH 2 – A model to predict County Sales

```
last = "2020-08-01"
model_county = smf.ols('County ~ Year + C(Month)',
                       data = County.loc[start:last,]).fit()
```

	Month	Year	County	County_pred
2020-04-01	April	2020	58.2	NaN
2020-05-01	May	2020	60.0	NaN
2020-06-01	June	2020	57.0	NaN
2020-07-01	July	2020	57.6	NaN
2020-08-01	August	2020	61.8	NaN

APPROACH 2 – A model to predict County Sales

```
last = "2020-08-01"
model_county = smf.ols('County ~ Year + C(Month)',
                       data = County.loc[start:last,]).fit()

County['County_pred'] = model_county.predict(\
                          County.loc["2020-09-01":,])
County.tail(9)
```

	Month	Year	County	County_pred
2020-04-01	April	2020	58.2	NaN
2020-05-01	May	2020	60.0	NaN
2020-06-01	June	2020	57.0	NaN
2020-07-01	July	2020	57.6	NaN
2020-08-01	August	2020	61.8	NaN
2020-09-01	September	2020	69.0	49.8875
2020-10-01	October	2020	75.0	51.9875
2020-11-01	November	2020	85.2	66.2375
2020-12-01	December	2020	121.8	105.6875

APPROACH 2 – A model to predict County Sales

```
last = "2020-08-01"
model_county = smf.ols('County ~ Year + C(Month)',
                       data = County.loc[start:last,]).fit()

County['County_pred'] = model_county.predict(\
                            County.loc["2020-09-01":,])
```

```
model_county.summary()
```

OLS Regression Results

Dep. Variable:	County	**R-squared:**	0.969
Model:	OLS	**Adj. R-squared:**	0.959
Method:	Least Squares	**F-statistic:**	92.02
Date:	Wed, 27 Mar 2024	**Prob (F-statistic):**	8.72e-23
Time:	18:01:34	**Log-Likelihood:**	-116.99
No. Observations:	48	**AIC:**	260.0

APPROACH 2 – Estimate County Excess Ratios

.

```
County['Excess_ratio'] = County['County']/County['County_pred']
County.tail()
```

	Month	Year	County	County_pred	Excess_ratio
2020-08-01	August	2020	61.8	NaN	NaN
2020-09-01	September	2020	69.0	49.8875	1.383112
2020-10-01	October	2020	75.0	51.9875	1.442654
2020-11-01	November	2020	85.2	66.2375	1.286280
2020-12-01	December	2020	121.8	105.6875	1.152454

APPROACH 2 – Estimate County Excess Ratios

```
County['Excess_ratio'] = County['County']/County['County_pred']
County.tail()
```

	Month	Year	County	County_pred	Excess_ratio
2020-08-01	August	2020	61.8	NaN	NaN
2020-09-01	September	2020	69.0	49.8875	1.383112
2020-10-01	October	2020	75.0	51.9875	1.442654
2020-11-01	November	2020	85.2	66.2375	1.286280
2020-12-01	December	2020	121.8	105.6875	1.152454

```
County.Excess_ratio.mean()
```

```
1.316125271414449
```

```
# On average, due to hurricane, County stores sold 31.6% more
```

APPROACH 2 – Predict Carlson Sales (had there be no hurricane)

```
model_Carlson = smf.ols('Carlson ~ Year + C(Month)',
                        data = Carlson).fit()
```

	Month	Year	Carlson
2020-04-01	April	2020	2.48
2020-05-01	May	2020	2.73
2020-06-01	June	2020	2.37
2020-07-01	July	2020	2.31
2020-08-01	August	2020	2.23
2020-09-01	September	2020	NaN
2020-10-01	October	2020	NaN
2020-11-01	November	2020	NaN
2020-12-01	December	2020	NaN

APPROACH 2 – Predict Carlson Sales (had there be no hurricane)

```
model_Carlson = smf.ols('Carlson ~ Year + C(Month)',
                        data = Carlson).fit()

Carlson['Carlson_pred'] = model_Carlson.predict(Carlson)
Carlson.tail(9)
```

	Month	Year	Carlson	Carlson_pred
				regular sales
2020-04-01	April	2020	2.48	2.487500
2020-05-01	May	2020	2.73	2.702500
2020-06-01	June	2020	2.37	2.482500
2020-07-01	July	2020	2.31	2.477500
2020-08-01	August	2020	2.23	2.542500
2020-09-01	September	2020	NaN	2.230833
2020-10-01	October	2020	NaN	2.548333
2020-11-01	November	2020	NaN	3.108333
2020-12-01	December	2020	NaN	4.520833

APPROACH 2 – Predict Carlson Sales (had there be no hurricane)

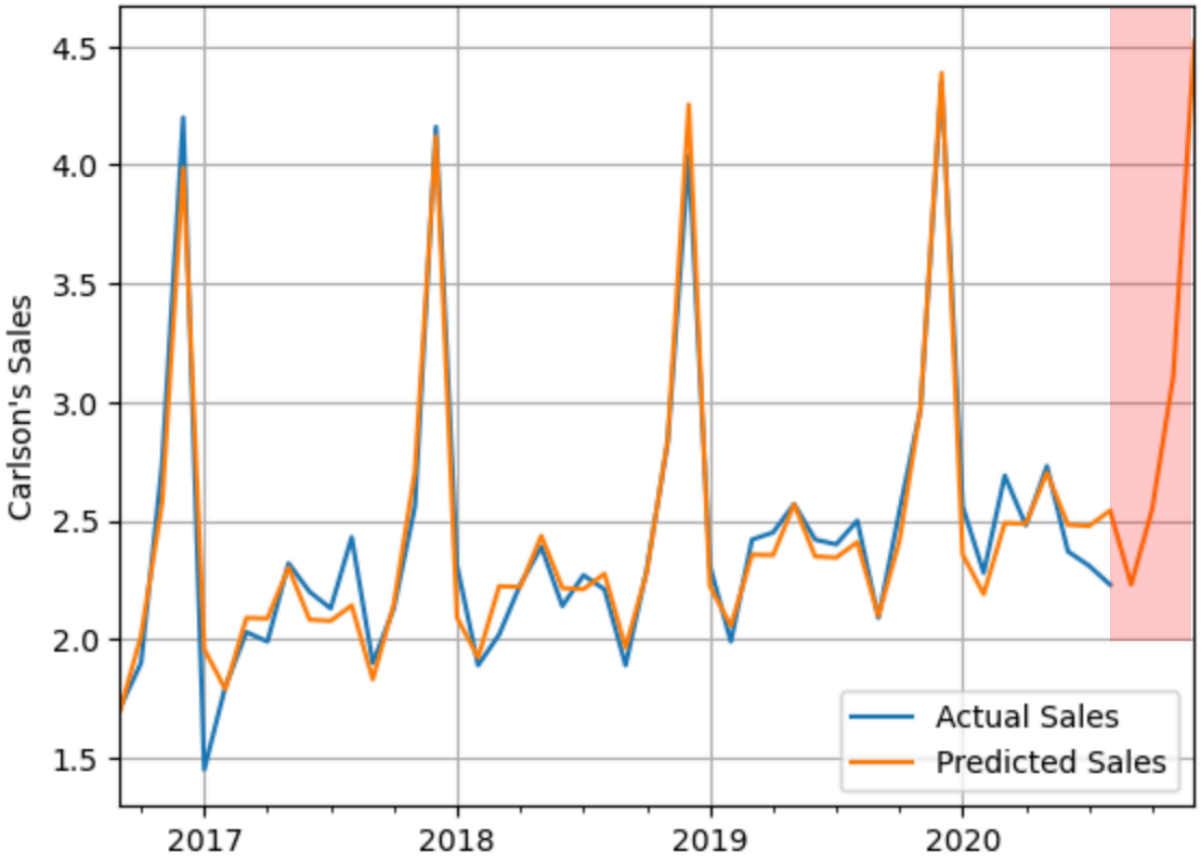
```
model_Carlson = smf.ols('Carlson ~ Year + C(Month)',
                        data = Carlson).fit()

Carlson['Carlson_pred'] = model_Carlson.predict(Carlson)
Carlson.tail(9)
```

	Month	Year	Carlson (plot)	Carlson_pred (plot)
2020-04-01	April	2020	2.48	2.487500
2020-05-01	May	2020	2.73	2.702500
2020-06-01	June	2020	2.37	2.482500
2020-07-01	July	2020	2.31	2.477500
2020-08-01	August	2020	2.23	2.542500
2020-09-01	September	2020	NaN	2.230833
2020-10-01	October	2020	NaN	2.548333
2020-11-01	November	2020	NaN	3.108333
2020-12-01	December	2020	NaN	4.520833

APPROACH 2 – Plot predicted Carlson Sales (had there be no hurricane)

```
list2 = ['Carlson','Carlson_pred']
Carlson[list2].plot()
plt.xlabel("")
plt.ylabel("Carlson's Sales")
legend_labels = ['Actual Sales','Predicted Sales']
plt.legend(labels = legend_labels)
```

Carlson	Carlson_pred
2.48	2.487500
2.73	2.702500
2.37	2.482500
2.31	2.477500
2.23	2.542500
NaN	2.230833
NaN	2.548333
NaN	3.108333
NaN	4.520833



APPROACH 2 – Model to predict Carlson's store Sales

```
model_Carlson.summary()
```

OLS Regression Results

Dep. Variable:	Carlson	**R-squared:**	0.947
Model:	OLS	**Adj. R-squared:**	0.929
Method:	Least Squares	**F-statistic:**	52.35
Date:	Wed, 27 Mar 2024	**Prob (F-statistic):**	1.01e-18
Time:	18:01:34	**Log-Likelihood:**	26.559
No. Observations:	48	**AIC:**	-27.12
Df Residuals:	35	**BIC:**	-2.792

APPROACH 2 – Carlson Store Lost Sales

```
df1 = County[['Excess_ratio']][-4:]
df1
```

	Excess_ratio
2020-09-01	1.383112
2020-10-01	1.442654
2020-11-01	1.286280
2020-12-01	1.152454

```
df2 = Carlson[['Carlson_pred']][-4:]
df2
```

	regular sales Carlson_pred
2020-09-01	2.230833
2020-10-01	2.548333
2020-11-01	3.108333
2020-12-01	4.520833

```
df3 = df1.join(df2)
```

	Excess_ratio	regular sales Carlson_pred
2020-09-01	1.383112	2.230833
2020-10-01	1.442654	2.548333
2020-11-01	1.286280	3.108333
2020-12-01	1.152454	4.520833

APPROACH 2 – Carlson Store Lost Sales

```
df1 = County[['Excess_ratio']][-4:]
df1
```

	Excess_ratio
2020-09-01	1.383112
2020-10-01	1.442654
2020-11-01	1.286280
2020-12-01	1.152454

```
df2 = Carlson[['Carlson_pred']][-4:]
df2
```

regular sales

	Carlson_pred
2020-09-01	2.230833
2020-10-01	2.548333
2020-11-01	3.108333
2020-12-01	4.520833

```
df3 = df1.join(df2)
df3['loss'] = df3['Excess_ratio']\
                * df3['Carlson_pred']
df3
```

regular sales

	Excess_ratio	Carlson_pred	loss
2020-09-01	1.383112	2.230833	3.085492
2020-10-01	1.442654	2.548333	3.676365
2020-11-01	1.286280	3.108333	3.998188
2020-12-01	1.152454	4.520833	5.210053

APPROACH 2 – Carlson Store Lost Sales

```
df1 = County[['Excess_ratio']][-4:]
df1
```

	Excess_ratio
2020-09-01	1.383112
2020-10-01	1.442654
2020-11-01	1.286280
2020-12-01	1.152454

```
df2 = Carlson[['Carlson_pred']][-4:]
df2
```

regular sales

	Carlson_pred
2020-09-01	2.230833
2020-10-01	2.548333
2020-11-01	3.108333
2020-12-01	4.520833

```
df3 = df1.join(df2)
df3['loss'] = df3['Excess_ratio']\
                * df3['Carlson_pred']
df3
```

regular sales

	Excess_ratio	Carlson_pred	loss
2020-09-01	1.383112	2.230833	3.085492
2020-10-01	1.442654	2.548333	3.676365
2020-11-01	1.286280	3.108333	3.998188
2020-12-01	1.152454	4.520833	5.210053

```
df3.loss.sum()
```

15.970098

Carlson Dept. Stores can claim a loss of $ 15,970,098