



FINANCIAL FORECASTING IN PYTHON

Building sensitive forecast models and common forecast assumptions

Victoria Clark

CGMA Financial Analyst



Considerations when forecasting

- Correctly interpret data
- Account for changes in data
- Account for interlinked variables
 - Dependencies
 - Sensitivities
- Set assumptions



Assumptions

- "Best guess" based on data available
- Set at the beginning of a forecast process
- Used to drive forecasting
- Can be directly controlled
- Can be indirectly controlled
 - Outside control of company



Different types of Assumptions

- Probability
 - Weighted
- Market sentiment
- Demand and supply





Working with pairs in Python

Using Combined Lists

Outcome	Probability (%)
1	30
2	20
3	50

```
outcome_probability = ['1|0.3', '2|0.2', '3|0.5']
```



Define a Python Function

Define a dependency or sensitivity formula

- Prevent duplication of work and errors

```
def assumption1()  
    if marketsentiment = 0.3:  
        sales + sales*0.1  
    else  
        sales
```



FINANCIAL FORECASTING IN PYTHON

Let's practice!



FINANCIAL FORECASTING IN PYTHON

Dependencies and sensitivity in financial forecasting

Victoria Clark

CGMA Financial Analyst



Explaining forecasting dependencies and sensitivities

- Interlinked variables
- Changing one variable has a knock-on effect on other variables





Working with dependencies and sensitivities in Python

```
if x == 0:  
    x_costs + y_costs  
else:  
    x_costs
```

- Expect rush orders
- Increases delivery costs by 10%

```
if month == December:  
    delivery_costs + delivery_costs*0.1  
else:  
    delivery_costs
```



FINANCIAL FORECASTING IN PYTHON

Let's practice!



FINANCIAL FORECASTING IN PYTHON

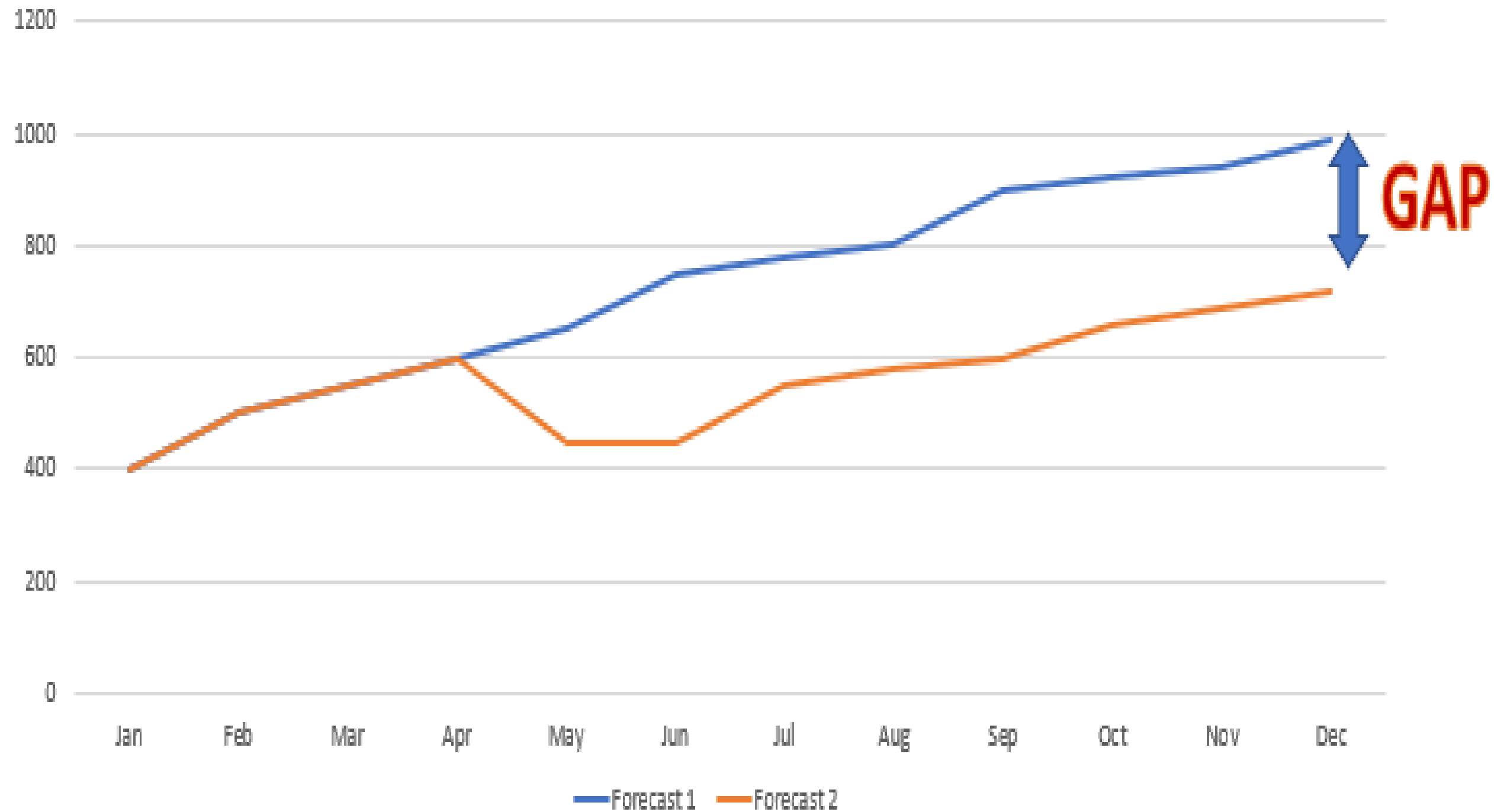
Working with variances in the forecast

Victoria Clark

CGMA Financial Analyst

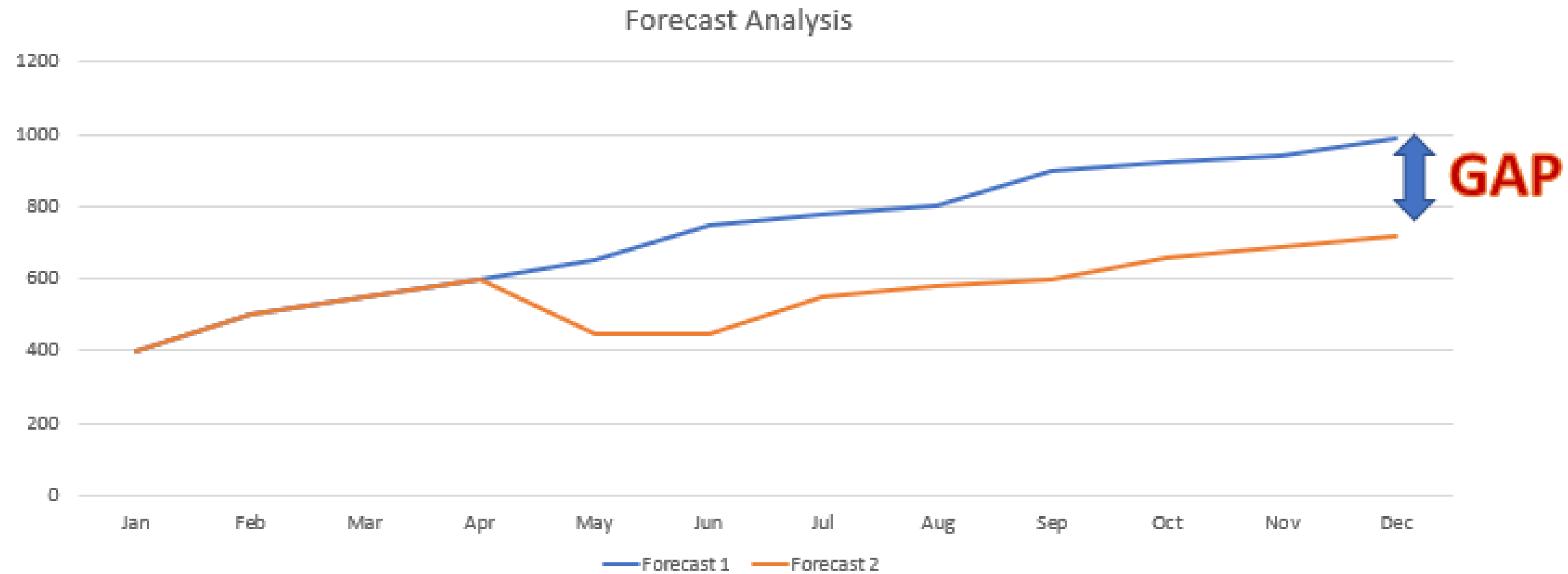


Forecast Analysis





A gap analysis



Gap analysis and alternative forecasts

```
rollingforecast1 = 1200
```

```
# First 6 months  
sales = 300
```

```
# The first dependency has 120 units  
dependency1 = 120
```

```
units = 30
```

```
expected_units = 45
```

```
# The adjusted dependency  
dependency2 = units + expected_units  
dependency
```

```
75
```



FINANCIAL FORECASTING IN PYTHON

Congratulations!