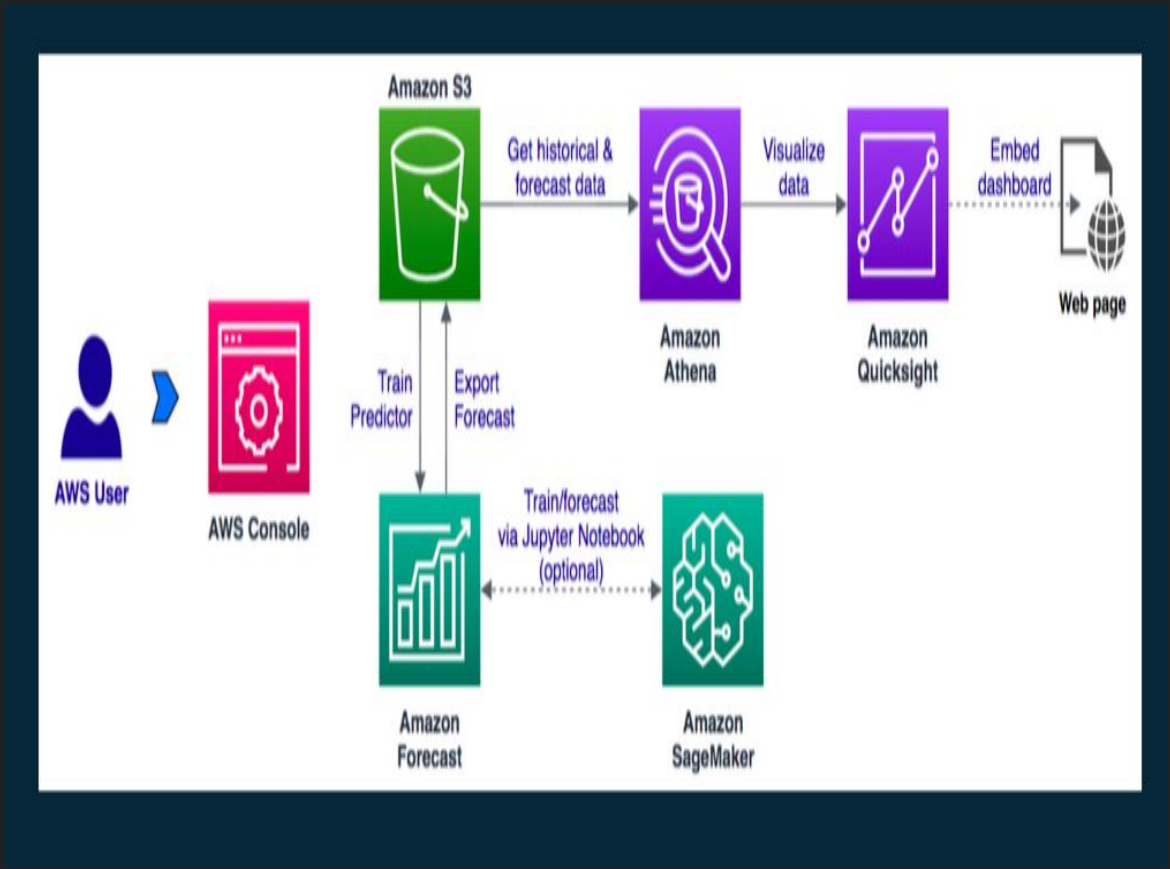
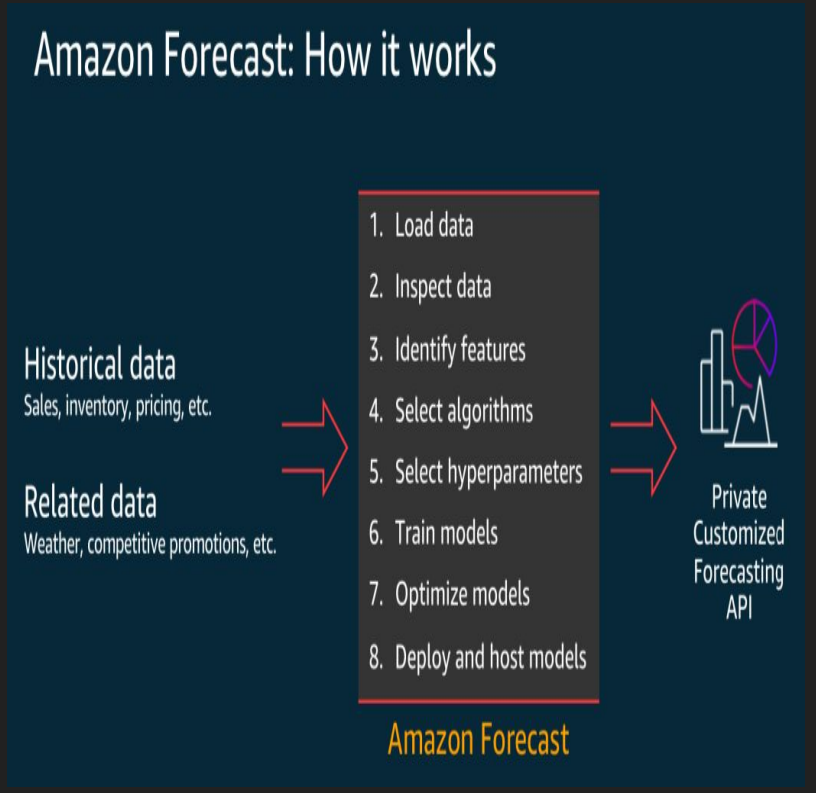
Console Demo Architecture





Traditional time-series models

- Independent forecasts
- Strong structural assumptions
- De facto industry standard
- Well-understood, >50 yrs. Research
- High data efficiency
- Data must match the structural assumptions
- Cannot identify patterns across time series

Deep learning time-series models

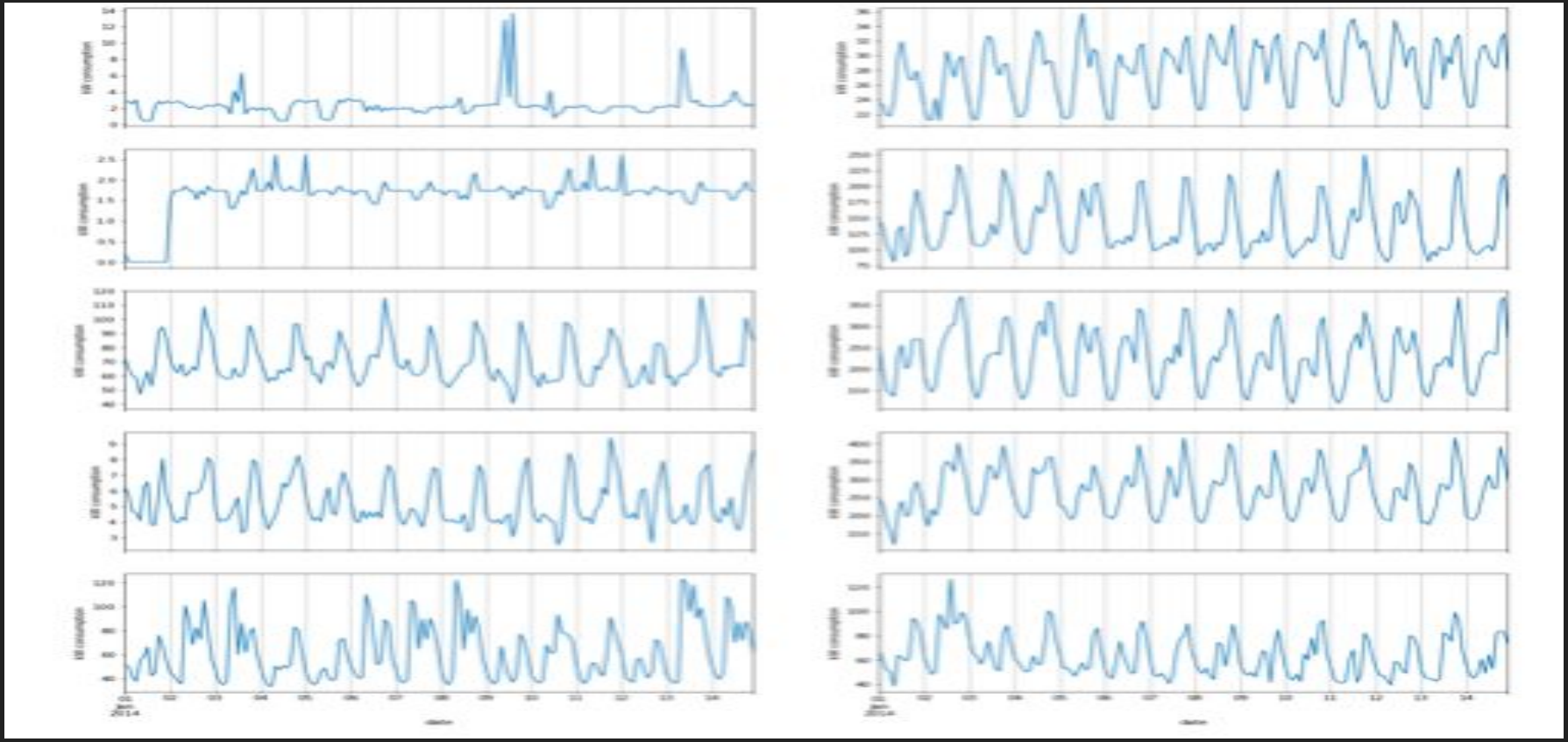
Global models: identify patterns using all available time series

- Group-dependent seasonality and lifecycle
- Behavior in response to covariate inputs
- Weak structural assumptions
- Can be significantly more accurate than traditional methods
- Can easily incorporate and learn from rich metadata
- Support cold-start forecasts for new items

Some Amazon Forecast Algorithms

- Auto-Regressive Integrated Moving Average (ARIMA)
- Exponential Smoothing (ETS)
- Non-Parametric Time Series (NPTS)
- Prophet
- Deep Auto-Regressive Plus (DeepAR+) – *Supports HPO*

Plot the resulting time series for the first ten customers for the time period spanning the first two weeks of 2014.



Perform predictions by deploying it to an endpoint

customer_id 119

forecast_day 76

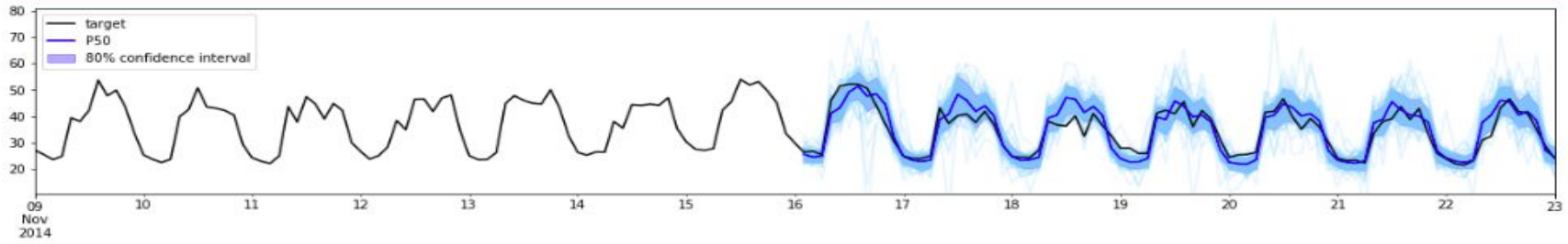
confidence 80

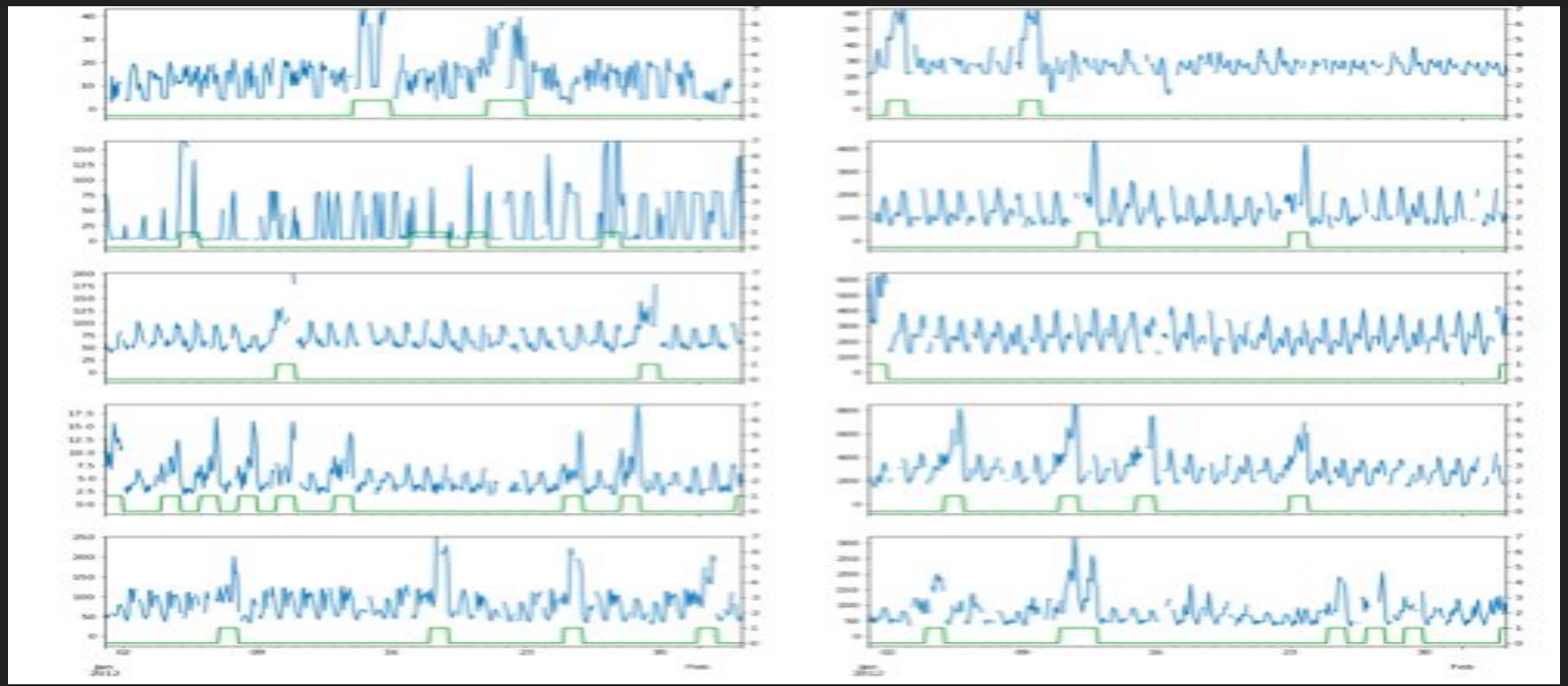
history_weeks_plot 1

show_samples

Run Interact

```
calling served model to generate predictions starting from 2014-11-16 00:00:00
/home/ec2-user/anaconda3/envs/mxnet_p36/lib/python3.6/site-packages/ipykernel/__main__.py:19: FutureWarning: Additi
on/subtraction of integers and integer-arrays to Timestamp is deprecated, will be removed in a future version.  Ins
tead of adding/subtracting `n`, use `n * self.freq`
/home/ec2-user/anaconda3/envs/mxnet_p36/lib/python3.6/site-packages/ipykernel/__main__.py:46: FutureWarning: Creati
ng a DatetimeIndex by passing range endpoints is deprecated.  Use `pandas.date_range` instead.
/home/ec2-user/anaconda3/envs/mxnet_p36/lib/python3.6/site-packages/ipykernel/__main__.py:48: FutureWarning: Additi
on/subtraction of integers and integer-arrays to Timestamp is deprecated, will be removed in a future version.  Ins
tead of adding/subtracting `n`, use `n * self.freq`
```





Perform predictions by deploying it to an endpoint