```python
data_preprocessor = DataPreprocessor()
df = data_preprocessor.load_kaggle_dataset()
df_full = data_preprocessor.create_full_date_index()
data_preprocessor.visualise_sales()

fe = FeatureEngineer(df_full)
df = fe.create_lag_features(lags=np.arange(1, 36, 2))
# fe.create_temporal_features()
data_dict = fe.create_X_y()
data_dict.keys()
```

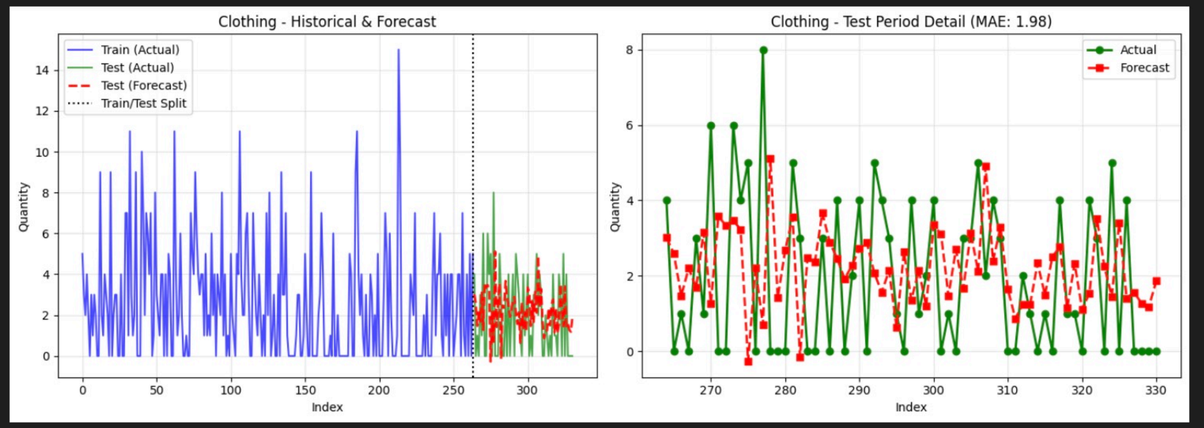
Run Cell | Run Above | Debug Cell

```python
# %% baseline time benchmark
start_time = time.time()

for category in data_dict.keys():
    category, result, test_mae = forecast_one_item(category,data_dict)

end_time = time.time()
print_time(
    "Total time taken (without parallel)",
    end_time - start_time,
    colour="yellow",
)
```

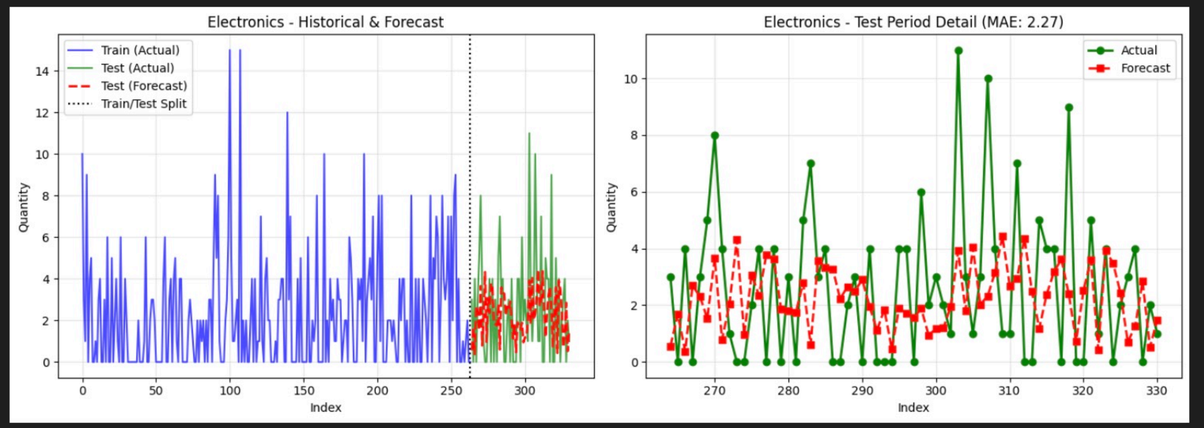
Run Cell | Run Above | Debug Cell

```python
# %% Use Ray for comparison
import ray
ray.init(ignore_reinit_error=True, logging_level='ERROR', log_to_driver=False
         )
start_time = time.time()
futures = [forecast_one_item_ray.remote(category, data_dict)
           for category in data_dict.keys()]
results = ray.get(futures)
for category, result_df, mae in results:
    print(f'{category} completed with MAE: {mae:.2f}')
end_time = time.time()
print_time(
    "Total time taken (with parallel)",
    end_time - start_time,
    colour="yellow",
)
ray.shutdown()
```

```
Beauty mae is 2.09
```




```
Clothing mae is 1.98
```




```
Electronics mae is 2.27

========== Total time taken (without parallel): 30.30 seconds ==========

Beauty completed with MAE: 2.09
Clothing completed with MAE: 1.98
Electronics completed with MAE: 2.27

========== Total time taken (with parallel): 10.71 seconds ==========
```