

2024

# OIBC Challenge

---

Team GEO

포항공과대학교 산업경영공학과 서재원

[seojw115@postech.ac.kr](mailto:seojw115@postech.ac.kr)

# 목차

01 제주 전력시장 및 SMP 가격

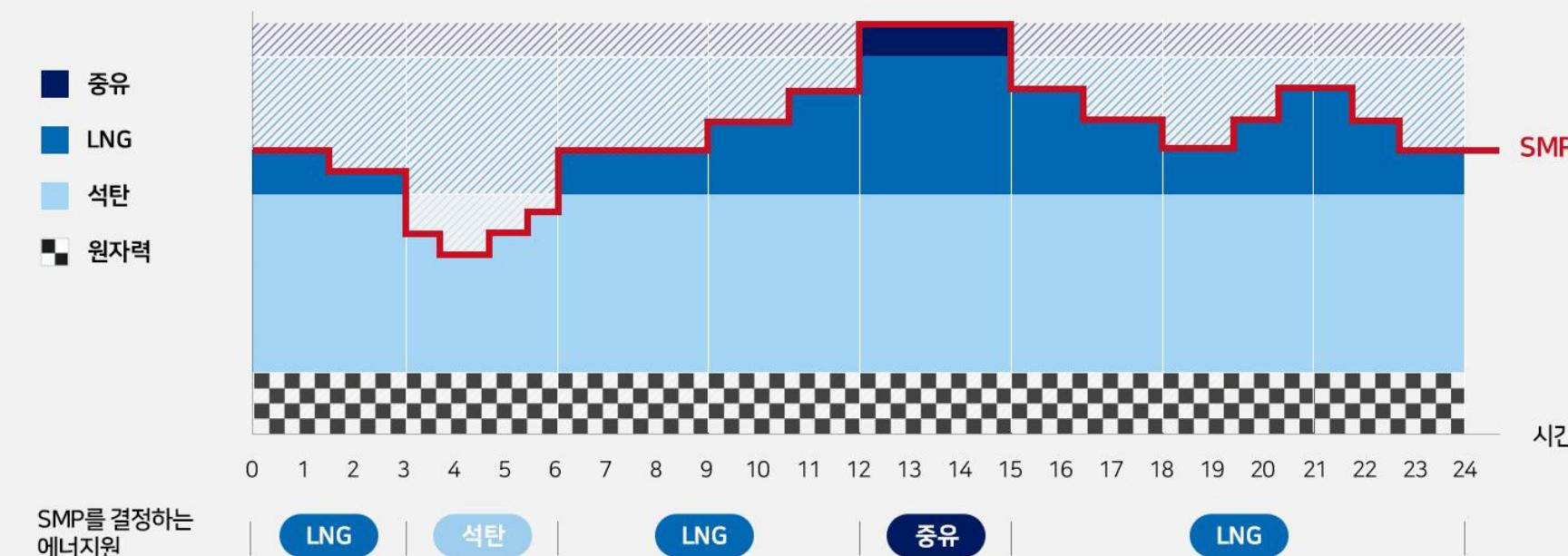
02 Data Preprocessing

03 Main Model

04 Model 개선 방향

## 제주 전력시장 및 SMP 가격

- SMP 가격: 전력 수요와 공급이 일치하는 시점의 발전비용이 가장 높은 발전방식의 변동비용
- 음의 SMP 가격: 날씨 등의 요인으로 인해 전력 수요 이상의 발전량이 공급될 때 형성
- 제주 전력시장: 전력시장 제도개선 시범사업의 일환으로 하루전시장, 실시간 시장으로 나누어 전력 시장을 운영



---

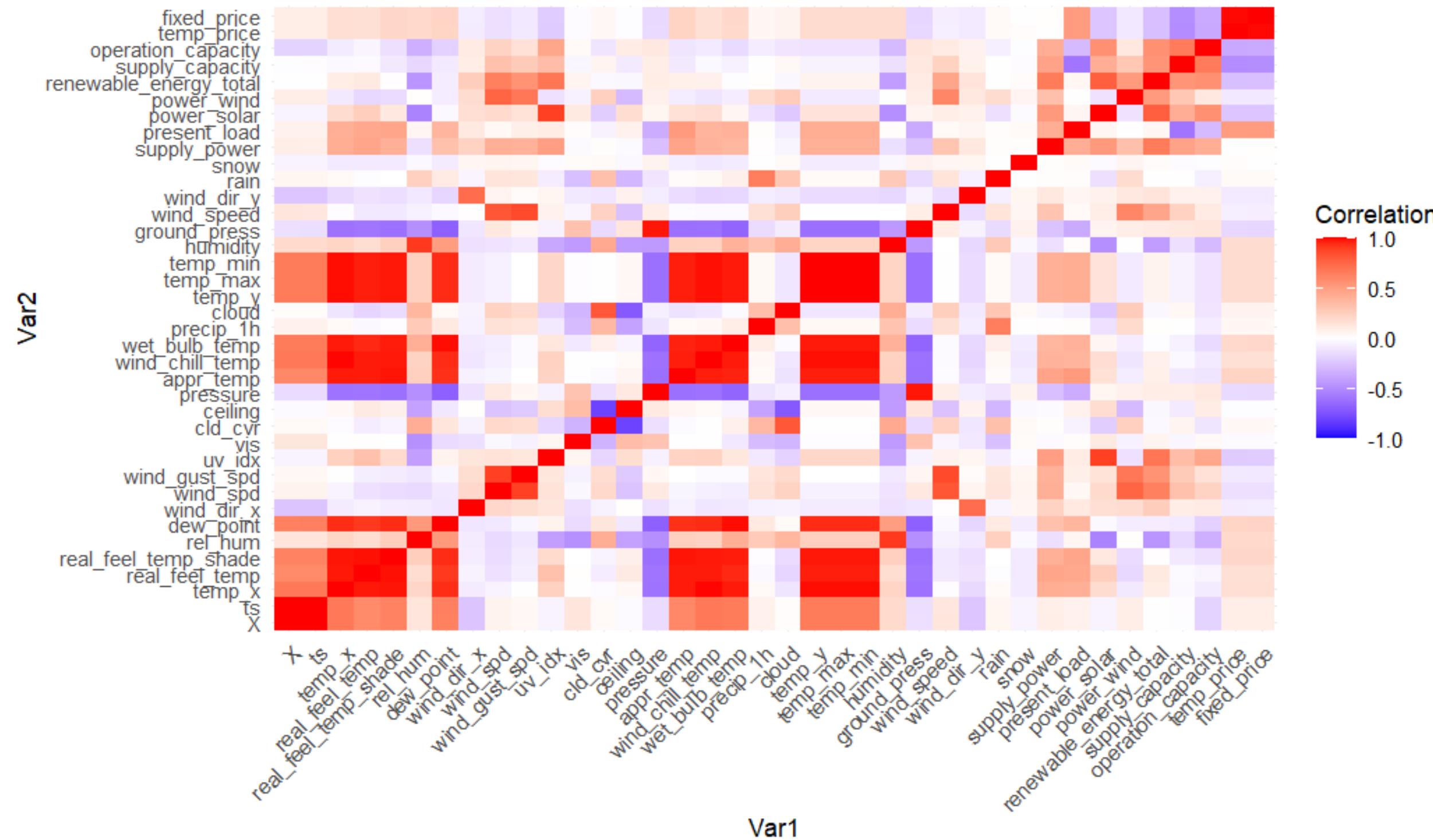
## Data Preprocessing

- 사용 데이터
  - 제주 전력시장 시장 전기가격: 하루전가격(smp\_da)
  - 제주 전력시장 현황 데이터: 공급능력, 현재수요, 태양광 발전량 등을 포함한 7개 columns
  - 기상 관측소 1 실측 데이터: temp, real\_feel\_temp, rel\_hum 등을 포함한 19개 columns
  - 기상 관측소 2 실측 데이터: cloud, wind\_speed, rain 등을 포함한 12개 columns

## Data Preprocessing

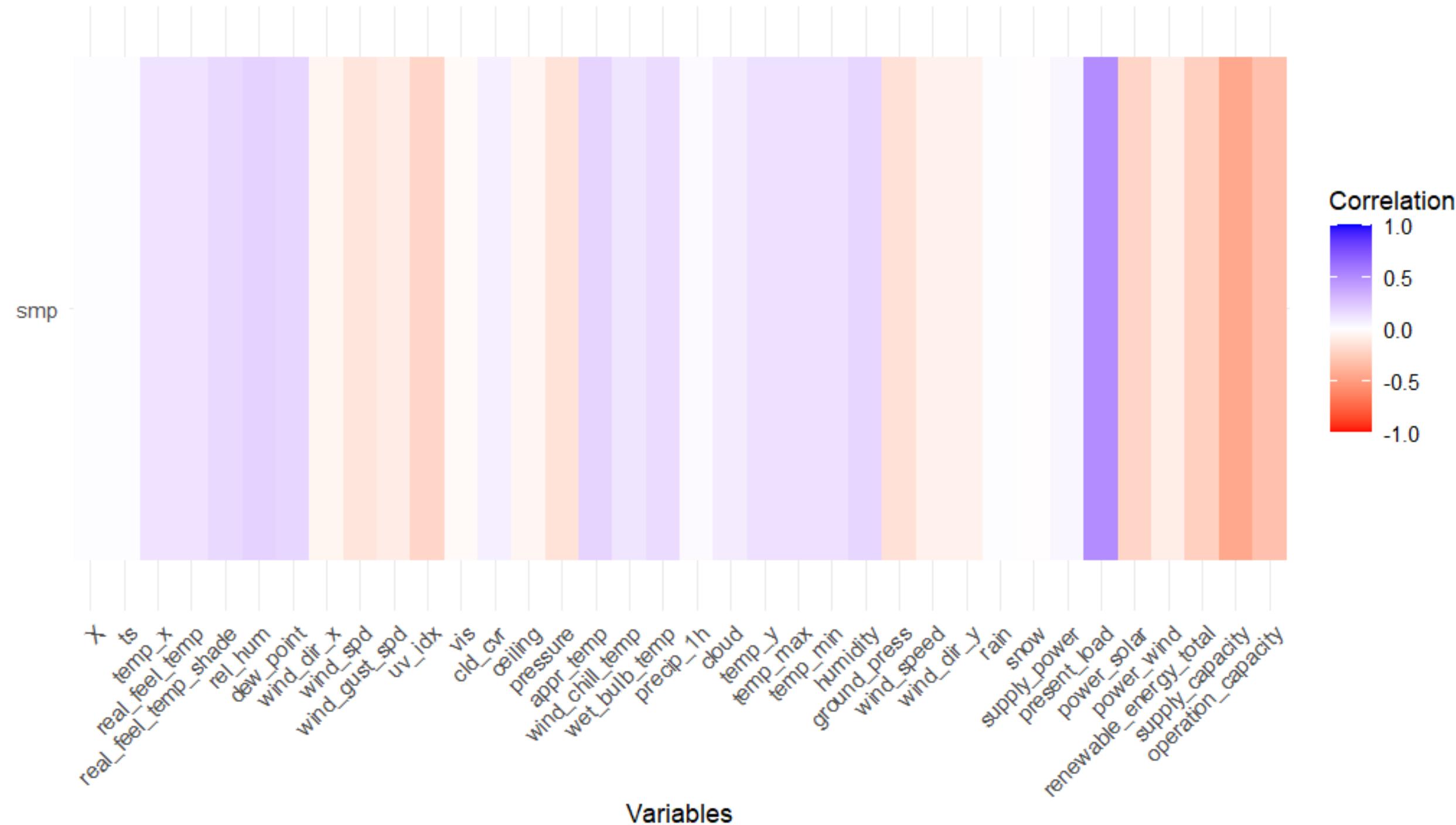
- SMP Price: 1시간 간격 제공, Weather Data: 15 분 간격 제공, 전력시장 현황: 5분 간격 제공  
→ ts 기준 1시간 간격으로 조정 후 평균값 도출
- 대회 기간 예측값 입력 시간이 전날 00시 ~ 11시였고, 기상실측데이터와 SMP 가격 호출 시간 사이에 딜레이가 발생해 48시간 이전까지의 측정값으로 SMP를 예측함
- ex) 11월 20일 SMP 예측값 입력 기한: 11월 19일 11시  
→ 11월 18일 까지의 측정값으로 11월 20일의 SMP를 예측함
- 기상실측데이터의 location은 무시 후 ts에 따라 평균값 도출
- Data normalize

# Data Preprocessing



# Data Preprocessing

Heatmap of Correlations with Fixed Price



## Data Preprocessing

- SMP Price와 낮은 상관관계를 보이는( $|corr| < 0.1$ ) 변수는 사용하지 않음
- 이후 변수간 높은 상관관계를 보이는 변수에 대해 VIF를 측정해 제거를 시도( $VIF > 10$ )
- 선불리 특정 변수를 제거하고 모델을 학습시키기 보다는 여러 변수 조합에 대한 모델 생성 후 가장 우수한 성능을 보이는 것을 선택
- "wind\_spd", "uv\_idx", "cld\_cvr", "ceiling", "pressure", "wind\_dir\_y", "rain", "snow", "supply\_power", "power\_wind"

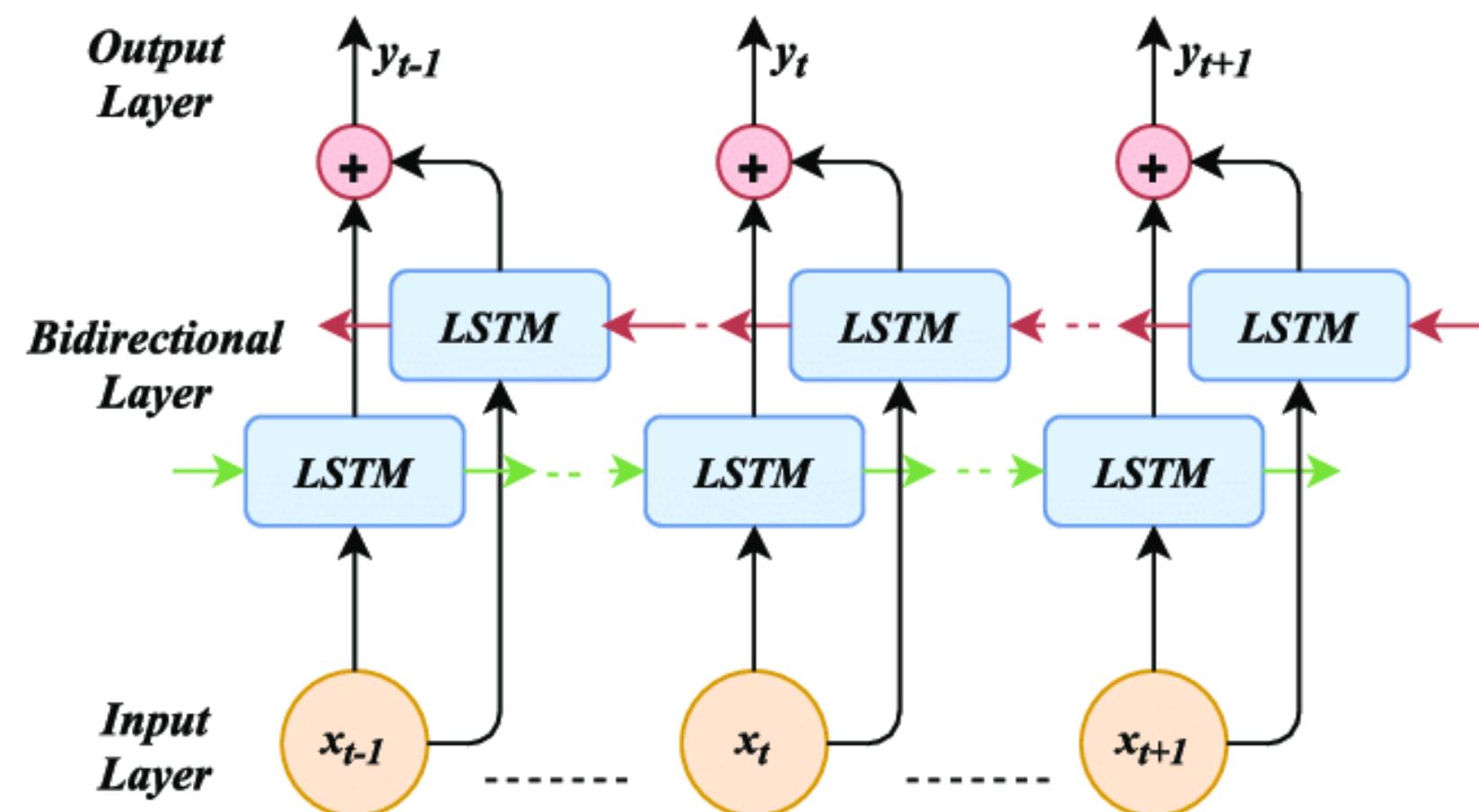
# Main Model

```
def set_model(self):
    self.model = Sequential([
        # 1차원 feature map 생성
        Conv1D(filters=64, kernel_size=7,
               padding="causal",
               activation = "leaky_relu",
               input_shape=[self.x_train_reshaped.shape[1], self.x_train_reshaped.shape[2]]),
        # LSTM
        Bidirectional(LSTM(32, return_sequences=True)),
        Dropout(0.2),
        Bidirectional(LSTM(16)),
        Dropout(0.2),
        Dense(16, activation="relu"),
        Dense(1),
    ])

    self.loss = Huber()
    self.optimizer = Adam(0.0005)
    self.model.compile(loss = Huber(), optimizer = self.optimizer, metrics = ["mse"])

    self.earlystopping = EarlyStopping(monitor = "val_loss", patience = 10)
    self.filename = os.path.join("tmp", "checkpointer.ckpt.weights.h5")
    self.checkpoint = ModelCheckpoint(self.filename, save_weights_only = True, save_best_only = True, monitor = "val_loss", verbose = 1)
```

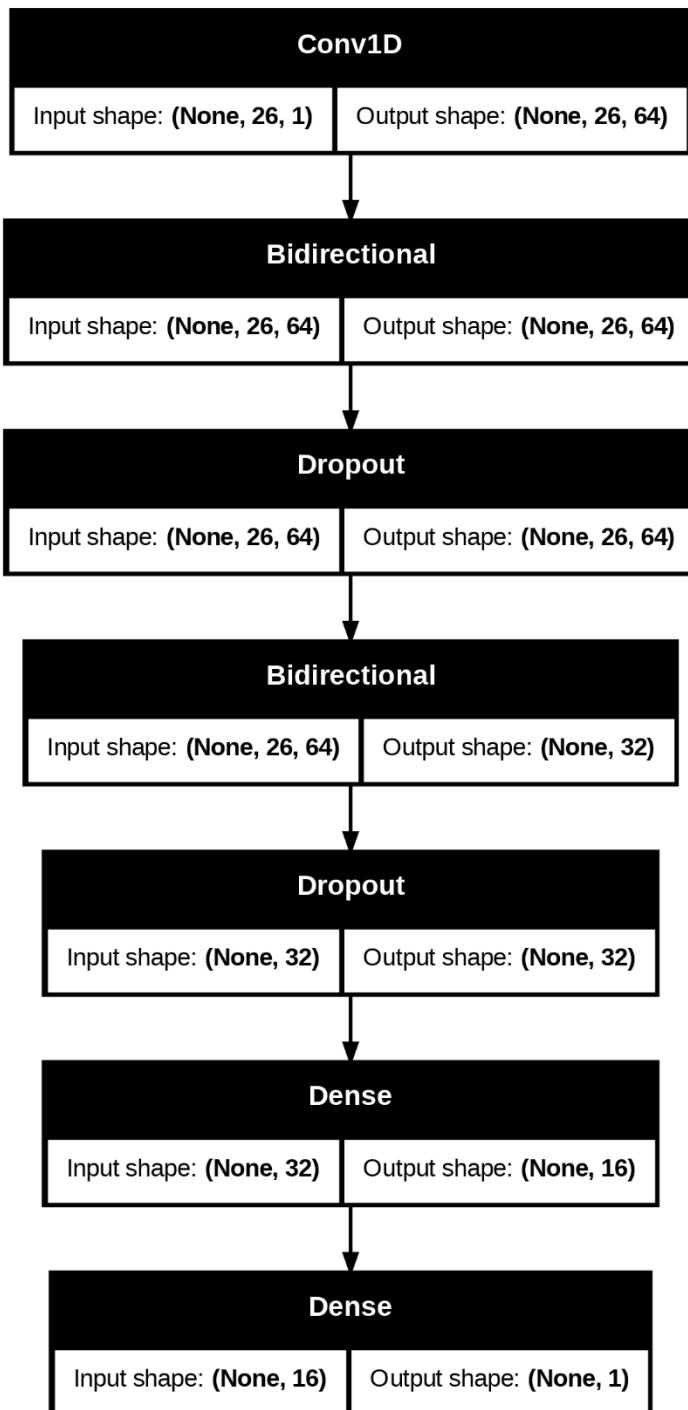
# Main Model



## Bidirectional LSTM

- 양방향으로 입력 시퀀스를 처리하기 때문에 전후 문맥 파악에 도움  
→ 전력 요금 예측시 전후 요금을 바탕으로 현재 요금 결정 가능
- 장기 의존성을 반영해 후 문맥 파악이 필요한 시계열 형식의 데이터에 효과적

# Main Model

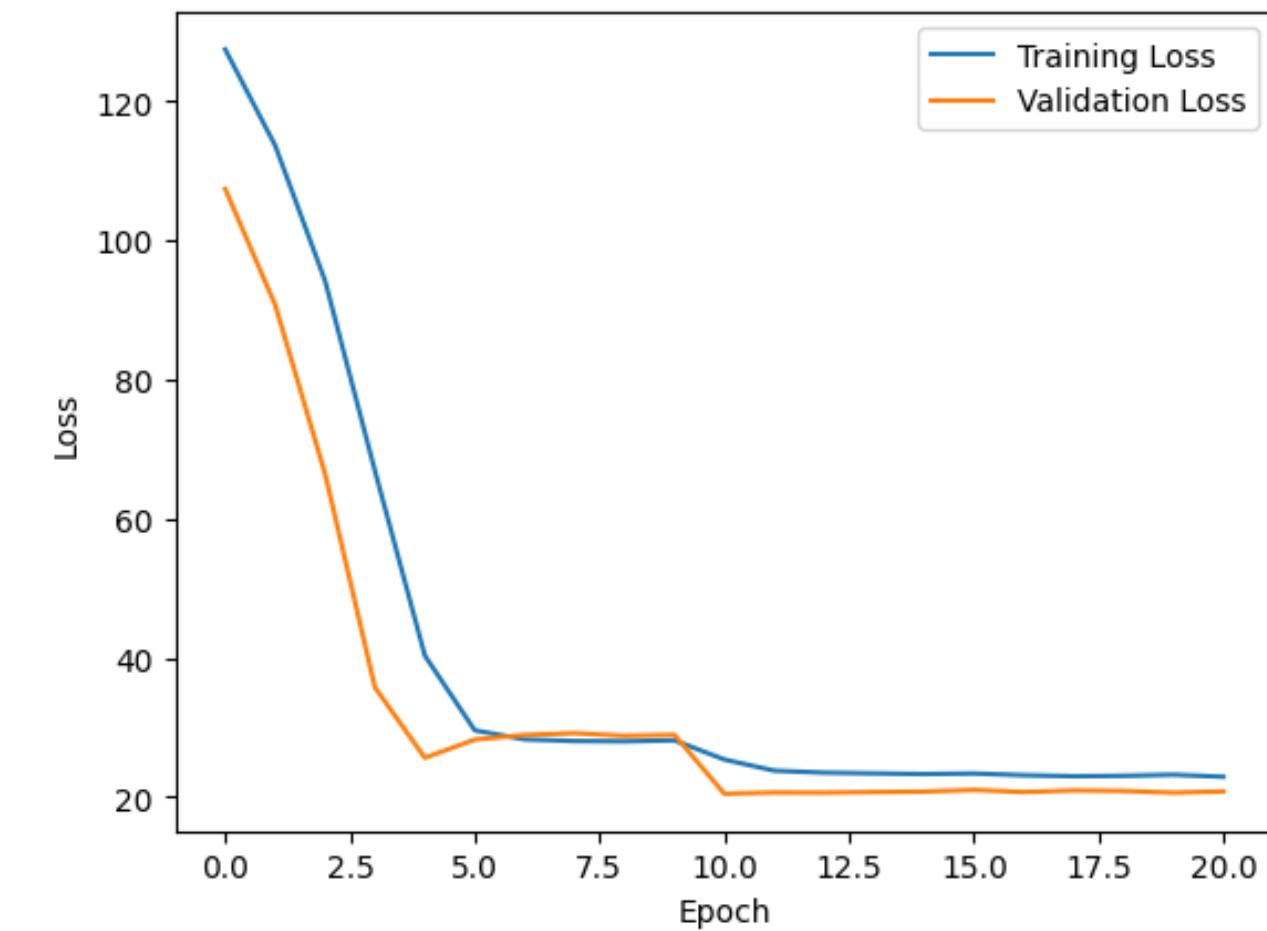


## Layers

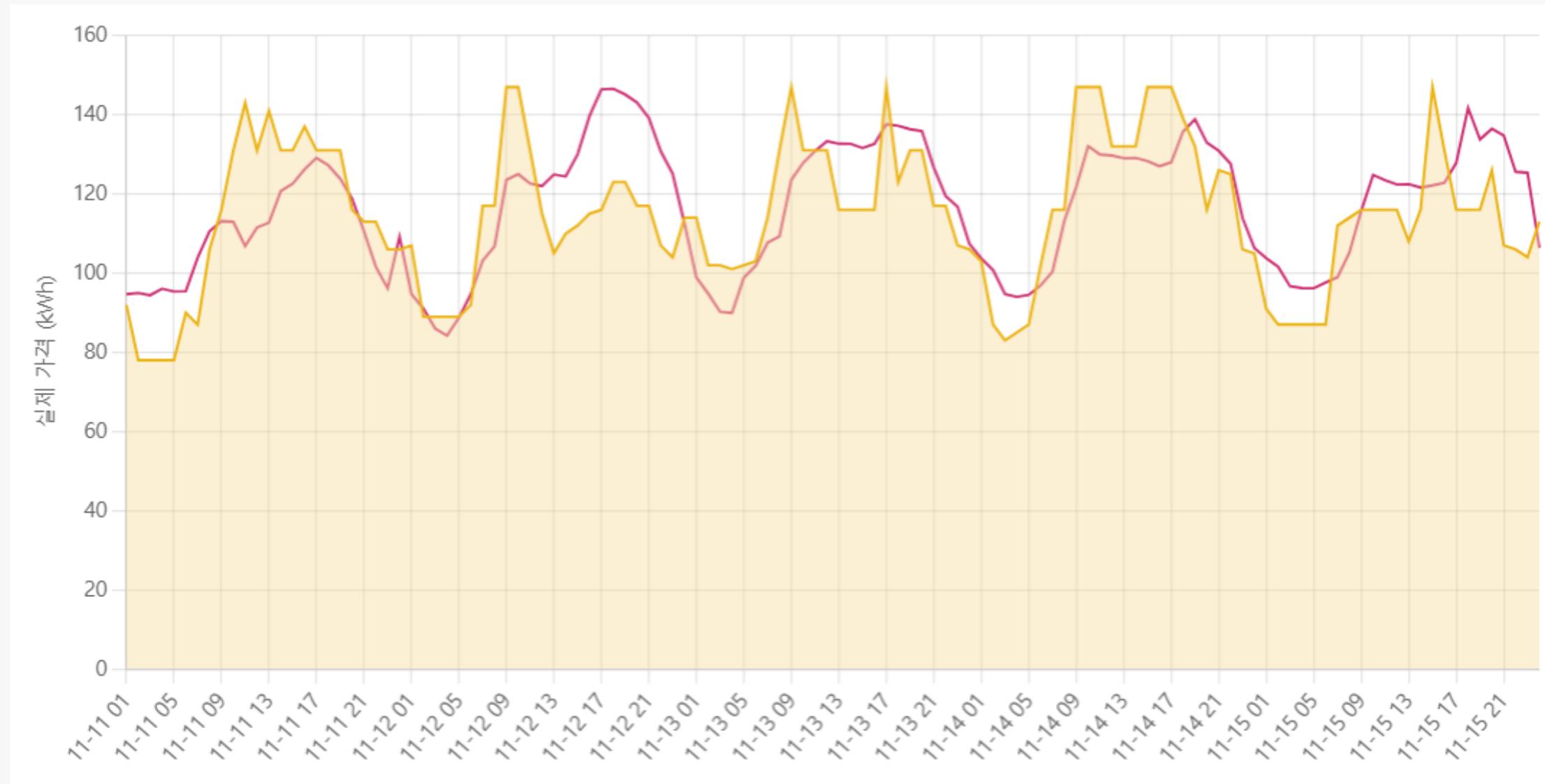
- Conv1D: 시계열 데이터의 특징 추출
- Bidirectional LSTM: 양방향 장기 의존성 반영
- Dropout: 과적합 방지, 일부 뉴런 비활성화
- Dense: 예측 결과 도출

# Main Model

Scaling 방식	Variable 선택 방법	LSTM Layer
Normalize	전부 사용	64
Min-Max	낮은 correlation 제거	32
	낮은 correlation & high VIF 제거	16



## Main Model



예측 오차율: -0.0297

---

## Model 개선 방향

- 음의 전력 요금을 예측 하는 방법의 구체화
  - 음의 전력 요금이 신재생 에너지 발전량과 밀접한 관계가 있다고 판단해 발전량 지표와 전력 요금의 관계를 파악해보려 했으나 실패함
  - 6000건의 ts에 대한 SMP 중 negative SMP는 300건 정도로 일반화를 하기엔 데이터가 부족했음
- 과소 계상되는 예측값에 대한 처리
  - 평가 기간 중 09~11시에 대한 예측값이 과소 계상되는 경향성을 보임
  - 가격 변동에 더 잘 반응하는 모델 구상 필요