

```

import pandas as pd
weather = pd.read_csv("WeatherPrediction.csv", index_col="DATE")

```

weather

	STATION	NAME	ACMH	ACSH	AWND	FMTM	PGTM	PRCP	SNOW	SNWD	...	WT13	WT14	WT15	WT16	WT17	WT18	WT19
DATE																		
01-01-1970	USW00094789	JFK INTERNATIONAL AIRPORT, NY US	80.0	90.0	NaN	NaN	NaN	0.00	0.0	0.0	...	NaN						
02-01-1970	USW00094789	JFK INTERNATIONAL AIRPORT, NY US	30.0	20.0	NaN	NaN	NaN	0.00	0.0	0.0	...	NaN						
03-01-1970	USW00094789	JFK INTERNATIONAL AIRPORT, NY US	80.0	100.0	NaN	NaN	NaN	0.02	0.0	0.0	...	NaN	NaN	NaN	1.0	NaN	1.0	NaN
04-01-1970	USW00094789	JFK INTERNATIONAL AIRPORT, NY US	10.0	20.0	NaN	NaN	NaN	0.00	0.0	0.0	...	NaN	NaN	NaN	NaN	NaN	1.0	NaN
05-01-1970	USW00094789	JFK INTERNATIONAL AIRPORT, NY US	30.0	10.0	NaN	NaN	NaN	0.00	0.0	0.0	...	NaN						
...
10-11-2023	USW00094789	JFK INTERNATIONAL AIRPORT, NY US	NaN	NaN	10.74	NaN	NaN	0.00	0.0	0.0	...	NaN						
11-11-2023	USW00094789	JFK INTERNATIONAL AIRPORT, NY US	NaN	NaN	10.74	NaN	NaN	0.00	0.0	0.0	...	NaN						
12-11-2023	USW00094789	JFK INTERNATIONAL AIRPORT, NY US	NaN	NaN	7.38	NaN	NaN	0.00	0.0	0.0	...	NaN						
13-11-2023	USW00094789	JFK INTERNATIONAL AIRPORT, NY US	NaN	NaN	6.04	NaN	2243.0	0.00	0.0	0.0	...	NaN						
14-11-2023	USW00094789	JFK INTERNATIONAL AIRPORT, NY US	NaN	NaN	NaN	NaN	NaN	0.00	0.0	0.0	...	NaN						

19676 rows × 45 columns

null pct gives the percentage of null values

```

null_pct = weather.apply(pd.isnull).sum()/weather.shape[0]
null_pct

```

STATION	0.000000
NAME	0.000000
ACMH	0.511334
ACSH	0.511283
AWND	0.260012
FMTM	0.485465
PGTM	0.376448

```
PRCP      0.000000
SNOW      0.000000
SNWD      0.000102
TAVG      0.666955
TMAX      0.000000
TMIN      0.000000
TSUN      0.998424
WDF1      0.511537
WDF2      0.488819
WDF5      0.493037
WDFG      0.739734
WDFM      0.999949
WESD      0.691452
WSF1      0.511384
WSF2      0.488819
WSF5      0.493088
WSFG      0.620705
WSFM      0.999949
WT01      0.630057
WT02      0.935099
WT03      0.932862
WT04      0.982568
WT05      0.981500
WT06      0.990801
WT07      0.994511
WT08      0.799502
WT09      0.992885
WT11      0.999288
WT13      0.888951
WT14      0.954920
WT15      0.997865
WT16      0.665735
WT17      0.996951
WT18      0.940689
WT19      0.999949
WT21      0.999746
WT22      0.997510
WV01      0.999949
dtype: float64
```

```
weather.apply(pd.isnull).sum()
```

```
STATION      0
NAME         0
ACMH        10061
ACSH        10060
AWND        5116
FMTM        9552
PGTM        7407
PRCP         0
SNOW         0
SNWD         2
TAVG       13123
TMAX         0
TMIN         0
TSUN       19645
WDF1        10065
WDF2        9618
WDF5        9701
WDFG        14555
WDFM        19675
WESD        13605
WSF1        10062
WSF2        9618
WSF5        9702
WSFG        12213
WSFM        19675
WT01        12397
WT02        18399
WT03        18355
WT04        19333
WT05        19312
WT06        19495
WT07        19568
WT08        15731
WT09        19536
WT11        19662
WT13        17491
WT14        18789
WT15        19634
```

```
WT16      13099  
WT17      19616  
WT18      18509  
WT19      19675  
WT21      19671  
WT22      19627  
WV01      19675  
dtype: int64
```

tells us how many columns are less than 50% null values

```
valid_columns = weather.columns[null_pct <.05]  
valid_columns  
Index(['STATION', 'NAME', 'PRCP', 'SNOW', 'SNWD', 'TMAX', 'TMIN'], dtype='object')
```

from next time it will only take valid_columns in use

```
weather = weather[valid_columns].copy()
```

```
weather.columns = weather.columns.str.lower()  
#to make all columns value in lower case
```

```
weather
```

	station		name	prcp	snow	snwd	tmax	tmin		
	DATE									
01-01-1970	USW00094789	JFK INTERNATIONAL AIRPORT, NY US		0.00	0.0	0.0	28	22		
02-01-1970	USW00094789	JFK INTERNATIONAL AIRPORT, NY US		0.00	0.0	0.0	31	22		
03-01-1970	USW00094789	JFK INTERNATIONAL AIRPORT, NY US		0.02	0.0	0.0	38	25		
04-01-1970	USW00094789	JFK INTERNATIONAL AIRPORT, NY US		0.00	0.0	0.0	31	23		
05-01-1970	USW00094789	JFK INTERNATIONAL AIRPORT, NY US		0.00	0.0	0.0	35	21		
...		
10-11-2023	USW00094789	JFK INTERNATIONAL AIRPORT, NY US		0.00	0.0	0.0	53	42		
11-11-2023	USW00094789	JFK INTERNATIONAL AIRPORT, NY US		0.00	0.0	0.0	56	39		
12-11-2023	USW00094789	JFK INTERNATIONAL AIRPORT, NY US		0.00	0.0	0.0	46	35		
13-11-2023	USW00094789	JFK INTERNATIONAL AIRPORT, NY US		0.00	0.0	0.0	48	31		
14-11-2023	USW00094789	JFK INTERNATIONAL AIRPORT, NY US		0.00	0.0	0.0	55	39		

19676 rows × 7 columns

```
weather = weather.ffill()
```

```
weather.apply(pd.isnull).sum()  
#to print all non null values
```

```
station    0  
name      0  
prcp      0  
snow      0  
snwd      0  
tmax      0  
tmin      0  
dtype: int64
```

```
weather.dtypes
```

```
station    object  
name      object  
prcp     float64  
snow     float64
```

```
snwd      float64
tmax      int64
tmin      int64
dtype: object

weather.index

Index(['01-01-1970', '02-01-1970', '03-01-1970', '04-01-1970', '05-01-1970',
       '06-01-1970', '07-01-1970', '08-01-1970', '09-01-1970', '10-01-1970',
       ...
       '05-11-2023', '06-11-2023', '07-11-2023', '08-11-2023', '09-11-2023',
       '10-11-2023', '11-11-2023', '12-11-2023', '13-11-2023', '14-11-2023'],
      dtype='object', name='DATE', length=19676)

weather.index = pd.to_datetime(weather.index)
#to convert object datetime into real date time

<ipython-input-100-c2252b39f1ba>:1: UserWarning: Parsing dates in DD/MM/YYYY format when dayfirst=False (the default) was specified.
weather.index = pd.to_datetime(weather.index)

weather.index

DatetimeIndex(['1970-01-01', '1970-02-01', '1970-03-01', '1970-04-01',
               '1970-05-01', '1970-06-01', '1970-07-01', '1970-08-01',
               '1970-09-01', '1970-10-01',
               ...
               '2023-05-11', '2023-06-11', '2023-07-11', '2023-08-11',
               '2023-09-11', '2023-10-11', '2023-11-11', '2023-12-11',
               '2023-11-13', '2023-11-14'],
              dtype='datetime64[ns]', name='DATE', length=19676, freq=None)

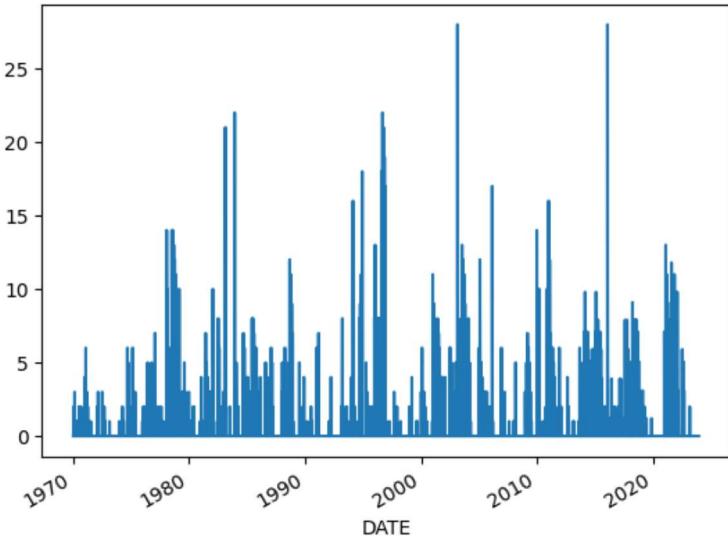
weather.index.year.value_counts().sort_index()
#to print years in order and to check if there is gap or not between data

1970    365
1971    365
1972    366
1973    365
1974    365
1975    365
1976    366
1977    365
1978    365
1979    365
1980    366
1981    365
1982    365
1983    365
1984    366
1985    365
1986    365
1987    365
1988    366
1989    365
1990    365
1991    365
1992    366
1993    365
1994    365
1995    365
1996    366
1997    365
1998    365
1999    365
2000    366
2001    365
2002    365
2003    365
2004    366
2005    365
2006    365
2007    365
2008    366
2009    365
2010    365
2011    365
```

```
2012    366
2013    365
2014    365
2015    365
2016    366
2017    365
2018    365
2019    365
2020    366
2021    365
2022    365
2023    318
Name: DATE, dtype: int64
```

```
weather["snwd"].plot()
#to check snow accumulation
```

```
<Axes: xlabel='DATE'>
```



```
weather
```

	station		name	prcp	snow	snwd	tmax	tmin	grid	bar
DATE										
1970-01-01	USW00094789	JFK INTERNATIONAL AIRPORT, NY US		0.00	0.0	0.0	28	22		
1970-02-01	USW00094789	JFK INTERNATIONAL AIRPORT, NY US		0.00	0.0	0.0	31	22		
1970-03-01	USW00094789	JFK INTERNATIONAL AIRPORT, NY US		0.02	0.0	0.0	38	25		
1970-04-01	USW00094789	JFK INTERNATIONAL AIRPORT, NY US		0.00	0.0	0.0	31	23		
1970-05-01	USW00094789	JFK INTERNATIONAL AIRPORT, NY US		0.00	0.0	0.0	35	21		
...		
2023-10-11	USW00094789	JFK INTERNATIONAL AIRPORT, NY US		0.00	0.0	0.0	53	42		
2023-11-11	USW00094789	JFK INTERNATIONAL AIRPORT, NY US		0.00	0.0	0.0	56	39		
2023-12-11	USW00094789	JFK INTERNATIONAL AIRPORT, NY US		0.00	0.0	0.0	46	35		
2023-11-13	USW00094789	JFK INTERNATIONAL AIRPORT, NY US		0.00	0.0	0.0	48	31		
2023-11-14	USW00094789	JFK INTERNATIONAL AIRPORT, NY US		0.00	0.0	0.0	55	39		

```
19676 rows × 7 columns
```

```
weather["target"] = weather.shift(-1)[ "tmax"]
#to predict tomorrow's data
```

```
weather
```

	station		name	prcp	snow	snwd	tmax	tmin	target	
DATE										
1970-01-01	USW00094789	JFK INTERNATIONAL AIRPORT, NY US		0.00	0.0	0.0	28	22	31.0	
1970-02-01	USW00094789	JFK INTERNATIONAL AIRPORT, NY US		0.00	0.0	0.0	31	22	38.0	
1970-03-01	USW00094789	JFK INTERNATIONAL AIRPORT, NY US		0.02	0.0	0.0	38	25	31.0	
1970-04-01	USW00094789	JFK INTERNATIONAL AIRPORT, NY US		0.00	0.0	0.0	31	23	35.0	
1970-05-01	USW00094789	JFK INTERNATIONAL AIRPORT, NY US		0.00	0.0	0.0	35	21	36.0	
...	
2023-10-11	USW00094789	JFK INTERNATIONAL AIRPORT, NY US		0.00	0.0	0.0	53	42	56.0	
2023-11-11	USW00094789	JFK INTERNATIONAL AIRPORT, NY US		0.00	0.0	0.0	56	39	46.0	
2023-12-11	USW00094789	JFK INTERNATIONAL AIRPORT, NY US		0.00	0.0	0.0	46	35	48.0	
2023-11-13	USW00094789	JFK INTERNATIONAL AIRPORT, NY US		0.00	0.0	0.0	48	31	55.0	
2023-11-14	USW00094789	JFK INTERNATIONAL AIRPORT, NY US		0.00	0.0	0.0	55	39	NaN	

19676 rows × 8 columns
weather = weather.fillna()
#to fill yesterday values bcz last columns last row value has a null value

weather

	station		name	prcp	snow	snwd	tmax	tmin	target	
DATE										
1970-01-01	USW00094789	JFK INTERNATIONAL AIRPORT, NY US		0.00	0.0	0.0	28	22	31.0	
1970-02-01	USW00094789	JFK INTERNATIONAL AIRPORT, NY US		0.00	0.0	0.0	31	22	38.0	
1970-03-01	USW00094789	JFK INTERNATIONAL AIRPORT, NY US		0.02	0.0	0.0	38	25	31.0	
1970-04-01	USW00094789	JFK INTERNATIONAL AIRPORT, NY US		0.00	0.0	0.0	31	23	35.0	
1970-05-01	USW00094789	JFK INTERNATIONAL AIRPORT, NY US		0.00	0.0	0.0	35	21	36.0	
...	
2023-10-11	USW00094789	JFK INTERNATIONAL AIRPORT, NY US		0.00	0.0	0.0	53	42	56.0	
2023-11-11	USW00094789	JFK INTERNATIONAL AIRPORT, NY US		0.00	0.0	0.0	56	39	46.0	
2023-12-11	USW00094789	JFK INTERNATIONAL AIRPORT, NY US		0.00	0.0	0.0	46	35	48.0	
2023-11-13	USW00094789	JFK INTERNATIONAL AIRPORT, NY US		0.00	0.0	0.0	48	31	55.0	
2023-11-14	USW00094789	JFK INTERNATIONAL AIRPORT, NY US		0.00	0.0	0.0	55	39	55.0	

19676 rows × 8 columns

```
from sklearn.linear_model import Ridge
rr = Ridge(alpha = .1)
#alpha control how much coefficient are shrunk for account with collinearity
```

```
weather.corr()
#to check correlation
```

```

<ipython-input-110-e30e408bed55>:1: FutureWarning: The default value of numeric_only in DataFrame.corr is deprecated. In a future
weather.corr()
    prcp      snow     snwd      tmax      tmin    target  [grid]
predictors = weather.columns[~weather.columns.isin(["target", "name", "station"])]
# to print values instead of target, name, and station

predictors

Index(['prcp', 'snow', 'snwd', 'tmax', 'tmin', 'target'])
       0.052093 -0.150510 -0.255152  0.955101  1.000000  0.915103

def backtest(weather, model, predictors, start=3650, step=90):
    all_predictions = []

    for i in range(start, weather.shape[0], step):
        train = weather.iloc[:i,:]
        test = weather.iloc[i:(i+step),:]

        model.fit(train[predictors], train["target"])

        preds = model.predict(test[predictors])
        preds = pd.Series(preds, index=test.index)
        combined = pd.concat([test["target"], preds], axis=1)
        combined.columns = ["actual", "prediction"]
        combined["diff"] = (combined["prediction"] - combined["actual"]).abs()

        all_predictions.append(combined)
    return pd.concat(all_predictions)

predictions = backtest(weather, rr, predictors)

predictions

```

	actual	prediction	diff	[grid]
DATE				[bar]
1979-12-30	43.0	50.229324	7.229324	
1979-12-31	42.0	43.673798	1.673798	
1980-01-01	41.0	41.579150	0.579150	
1980-02-01	36.0	43.961887	7.961887	
1980-03-01	30.0	40.204726	10.204726	
...	
2023-10-11	56.0	55.171591	0.828409	
2023-11-11	46.0	54.973136	8.973136	
2023-12-11	48.0	48.421409	0.421409	
2023-11-13	55.0	47.258643	7.741357	
2023-11-14	55.0	54.524056	0.475944	

16026 rows × 3 columns

```

from sklearn.metrics import mean_absolute_error
mean_absolute_error(predictions["actual"], predictions["prediction"])

5.135999231878063

predictions["diff"].mean()

5.135999231878063

```

```

def pct_diff(old,new):
    return (new - old)/old
def compute_rolling(weather,horizon,col):
    label = f"rolling_{horizon}_{col}"
    weather[label] = weather[col].rolling(horizon).mean()
    weather[f"{label}_pct"] = pct_diff(weather[label],weather[col])
    return weather
rolling_horizons = [3,14]
for horizon in rolling_horizons:
    for col in ["tmax","tmin","prcp"]:
        weather = compute_rolling(weather,horizon,col)

```

weather

		station	name	prcp	snow	snwd	tmax	tmin	target	rolling_3_tmax	rolling_3_tmax_pct	rolling_3_tmin	roll
DATE													
1970-01-01	USW00094789	JFK INTERNATIONAL AIRPORT, NY US		0.00	0.0	0.0	28	22	31.0	NaN	NaN	NaN	NaN
1970-02-01	USW00094789	JFK INTERNATIONAL AIRPORT, NY US		0.00	0.0	0.0	31	22	38.0	NaN	NaN	NaN	NaN
1970-03-01	USW00094789	JFK INTERNATIONAL AIRPORT, NY US		0.02	0.0	0.0	38	25	31.0	32.333333	0.175258	23.000000	
1970-04-01	USW00094789	JFK INTERNATIONAL AIRPORT, NY US		0.00	0.0	0.0	31	23	35.0	33.333333	-0.070000	23.333333	
1970-05-01	USW00094789	JFK INTERNATIONAL AIRPORT, NY US		0.00	0.0	0.0	35	21	36.0	34.666667	0.009615	23.000000	
...
2023-10-11	USW00094789	JFK INTERNATIONAL AIRPORT, NY US		0.00	0.0	0.0	53	42	56.0	55.000000	-0.036364	40.000000	
2023-11-11	USW00094789	JFK INTERNATIONAL AIRPORT, NY US		0.00	0.0	0.0	56	39	46.0	56.000000	0.000000	39.666667	
2023-12-11	USW00094789	JFK INTERNATIONAL AIRPORT, NY US		0.00	0.0	0.0	46	35	48.0	51.666667	-0.109677	38.666667	
2023-11-13	USW00094789	JFK INTERNATIONAL AIRPORT, NY US		0.00	0.0	0.0	48	31	55.0	50.000000	-0.040000	35.000000	
2023-11-14	USW00094789	JFK INTERNATIONAL AIRPORT, NY US		0.00	0.0	0.0	55	39	55.0	49.666667	0.107383	35.000000	

19676 rows × 20 columns

weather=weather.iloc[14:,:]

weather

	station	name	prcp	snow	snwd	tmax	tmin	target	rolling_3_tmax	rolling_3_tmax_pct	rolling_3_tmin	roll
DATE												
1970-01-15	USW00094789	JFK INTERNATIONAL AIRPORT, NY US	0.00	0.0	0.0	29	13	36.0	29.666667	-0.022472	18.000000	
1970-01-16	USW00094789	JFK INTERNATIONAL AIRPORT, NY US	0.00	0.0	0.0	36	21	43.0	30.333333	0.186813	16.666667	
1970-01-17	USW00094789	JFK INTERNATIONAL AIRPORT, NY US	0.02	0.0	0.0	43	30	42.0	36.000000	0.194444	21.333333	
1970-01-18	USW00094789	JFK INTERNATIONAL AIRPORT, NY US	0.10	0.0	0.0	42	25	25.0	40.333333	0.041322	25.333333	
1970-01-19	USW00094789	JFK INTERNATIONAL AIRPORT, NY US	0.00	0.0	0.0	25	16	24.0	36.666667	-0.318182	23.666667	
...	
2023-10-11	USW00094789	JFK INTERNATIONAL AIRPORT, NY US	0.00	0.0	0.0	53	42	56.0	55.000000	-0.036364	40.000000	
2023-11-11	USW00094789	JFK INTERNATIONAL AIRPORT, NY US	0.00	0.0	0.0	56	39	46.0	56.000000	0.000000	39.666667	
2023-12-11	USW00094789	JFK INTERNATIONAL AIRPORT, NY US	0.00	0.0	0.0	46	35	48.0	51.666667	-0.109677	38.666667	
2023-11-13	USW00094789	JFK INTERNATIONAL AIRPORT, NY US	0.00	0.0	0.0	48	31	55.0	50.000000	-0.040000	35.000000	
2023-11-14	USW00094789	JFK INTERNATIONAL AIRPORT, NY US	0.00	0.0	0.0	55	39	55.0	49.666667	0.107383	35.000000	

19662 rows × 20 columns

```
weather=weather.fillna(0)
#to fill null values by 0
```

```
def expand_mean(df):
    return df.expanding(1).mean()
for col in ["tmax","tmin","prcp"]:
    weather[f"month_avg{col}"] = weather[col].groupby(weather.index.month,group_keys=False).apply(expand_mean)
    weather[f"day_avg_{col}"] = weather[col].groupby(weather.index.day_of_year, group_keys=False).apply(expand_mean)
```

weather

	station	name	prcp	snow	snwd	tmax	tmin	target	rolling_3_tmax	rolling_3_tmax_pct	...	rolling_14_tmin
DATE												
1970-01-15	USW00094789	JFK INTERNATIONAL AIRPORT, NY US	0.00	0.0	0.0	29	13	36.0	29.666667	-0.022472	...	18.857143
1970-01-16	USW00094789	JFK INTERNATIONAL AIRPORT, NY US	0.00	0.0	0.0	36	21	43.0	30.333333	0.186813	...	18.785714
1970-01-17	USW00094789	JFK INTERNATIONAL AIRPORT, NY US	0.02	0.0	0.0	43	30	42.0	36.000000	0.194444	...	19.142857
1970-01-18	USW00094789	JFK INTERNATIONAL AIRPORT, NY US	0.10	0.0	0.0	42	25	25.0	40.333333	0.041322	...	19.285714
1970-01-19	USW00094789	JFK INTERNATIONAL AIRPORT, NY US	0.00	0.0	0.0	25	16	24.0	36.666667	-0.318182	...	18.928571
...
2023-10-11	USW00094789	JFK INTERNATIONAL AIRPORT, NY US	0.00	0.0	0.0	53	42	56.0	55.000000	-0.036364	...	44.214286
2023-11-11	USW00094789	JFK INTERNATIONAL AIRPORT, NY US	0.00	0.0	0.0	56	39	46.0	56.000000	0.000000	...	42.857143
2023-12-11	USW00094789	JFK INTERNATIONAL AIRPORT, NY US	0.00	0.0	0.0	46	35	48.0	51.666667	-0.109677	...	41.642857
2023-11-13	USW00094789	JFK INTERNATIONAL AIRPORT, NY US	0.00	0.0	0.0	48	31	55.0	50.000000	-0.040000	...	40.285714

```
predictors = weather.columns[~weather.columns.isin(["target", "name", "station"])]
```

```
2023-11-13 00:00:00+00:00 INTERNATIONAL 0.00 0.0 0.0 55 31 55.0 50.000000 -0.040000 ... 40.285714
```

```
predictors
```

```
Index(['prcp', 'snow', 'snwd', 'tmax', 'tmin', 'rolling_3_tmax',
       'rolling_3_tmax_pct', 'rolling_3_tmin', 'rolling_3_tmin_pct',
       'rolling_3_prcp', 'rolling_3_prcp_pct', 'rolling_14_tmax',
       'rolling_14_tmax_pct', 'rolling_14_tmin', 'rolling_14_tmin_pct',
       'rolling_14_prcp', 'rolling_14_prcp_pct', 'month_avgtmax',
       'day_avg_tmax', 'month_avgtmin', 'day_avg_tmin', 'month_avgprcp',
       'day_avg_prcp'],
      dtype='object')
```

```
predictions = backtest(weather, rr, predictors)
```

```
mean_absolute_error(predictions["actual"], predictions["prediction"])
```

```
4.852470795836304
```

```
predictions.sort_values("diff", ascending=False)
```

	actual	prediction	diff	
DATE				
1990-12-03	85.0	53.601832	31.398168	
1998-03-26	80.0	51.356424	28.643576	
2007-03-26	78.0	49.734100	28.265900	
2003-04-15	86.0	58.011974	27.988026	
1985-04-18	84.0	58.209347	25.790653	
...	
2023-02-10	75.0	74.997822	0.002178	
2021-04-08	81.0	81.001341	0.001341	

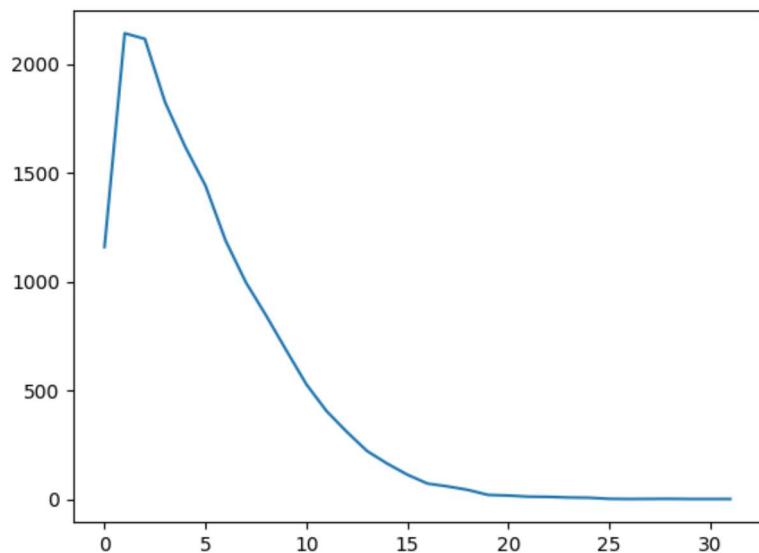
weather.loc["1990-03-07": "1990-03-17"]

	station		name	prcp	snow	snwd	tmax	tmin	target	rolling_3_tmax	rolling_3_tmax_pct	...	rolling_14_tmin
DATE													
1990-03-13	USW00094789	JFK INTERNATIONAL AIRPORT, NY US		0.00	0.0	0.0	85	41	62.0	67.666667	0.256158	...	29.500000
1990-03-14	USW00094789	JFK INTERNATIONAL AIRPORT, NY US		0.00	0.0	0.0	62	46	55.0	68.666667	-0.097087	...	30.857143
1990-03-15	USW00094789	JFK INTERNATIONAL AIRPORT, NY US		0.00	0.0	0.0	55	43	62.0	67.333333	-0.183168	...	32.214286
1990-03-16	USW00094789	JFK INTERNATIONAL AIRPORT, NY US		0.00	0.0	0.0	62	48	61.0	59.666667	0.039106	...	33.428571
1990-03-17	USW00094789	JFK INTERNATIONAL AIRPORT, NY US		0.26	0.0	0.0	61	49	59.0	59.333333	0.028090	...	34.357143
1990-03-07	USW00094789	JFK INTERNATIONAL AIRPORT, NY US		0.00	0.0	0.0	85	63	90.0	79.666667	0.066946	...	66.285714
1990-03-08	USW00094789	JFK INTERNATIONAL AIRPORT, NY US		0.00	0.0	0.0	89	72	85.0	86.666667	0.026923	...	70.071429
1990-03-09	USW00094789	JFK INTERNATIONAL AIRPORT, NY US		0.00	0.0	0.0	77	63	74.0	79.333333	-0.029412	...	67.071429
1990-03-10	USW00094789	JFK INTERNATIONAL AIRPORT, NY US		0.00	0.0	0.0	69	50	73.0	69.333333	-0.004808	...	56.071429
1990-03-11	USW00094789	JFK INTERNATIONAL AIRPORT, NY US		0.00	0.0	0.0	75	53	70.0	69.666667	0.076555	...	47.428571
1990-03-12	USW00094789	JFK INTERNATIONAL AIRPORT, NY US		0.85	0.0	0.0	53	38	58.0	53.666667	-0.012422	...	39.500000

11 rows × 26 columns

```
predictions["diff"].round().value_counts().sort_index().plot()
```

<Axes: >



predictions

	actual	prediction	diff	DATE
1980-01-13	54.0	32.702958	21.297042	
1980-01-14	51.0	45.058331	5.941669	
1980-01-15	45.0	46.978986	1.978986	
1980-01-16	40.0	42.440026	2.440026	
1980-01-17	41.0	41.161413	0.161413	
...	
2023-10-11	56.0	57.268985	1.268985	
2023-11-11	46.0	56.039327	10.039327	
2023-12-11	48.0	50.530314	2.530314	
2023-11-13	55.0	48.972861	6.027139	
2023-11-14	55.0	55.403393	0.403393	

16012 rows × 3 columns