negoita-ofrim-final-project

June 16, 2023

1 Predicting the apparent temperature based on other meteorological factors

1.0.1 Abstract

This research paper aims to explore the relationship between apparent temperature and various meteorological factors through the application of multiple regression analysis. The study focuses on determining the association between apparent temperature, actual temperature, air humidity, wind speed, wind bearing, visibility, and atmospheric pressure. By employing a comprehensive dataset, encompassing these variables over a specific period, our analysis aims to identify significant predictors contributing to changes in apparent temperature. The findings from this study could provide valuable insights into the complex interactions among meteorological factors, enabling better understanding and prediction of the apparent temperature, and ultimately contributing to improved decision-making in various domains such as public health, urban planning, and climate adaptation strategies.

[27]:

```python
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import statsmodels.api as sm
from scipy import stats

sns.set_style("whitegrid")
```

1.0.2 Introduction

Weather conditions have a profound impact on human comfort, health, and daily activities. Among the various weather-related factors, apparent temperature plays a crucial role in determining how individuals perceive and respond to the thermal environment. Apparent temperature, also known as the "feels like" temperature, takes into account not only the actual temperature but also other meteorological parameters such as air humidity, wind speed, wind bearing, visibility, and atmospheric pressure.

The purpose of this research is to investigate the relationship between apparent temperature and the aforementioned meteorological factors using multiple regression analysis. Multiple regression allows for the examination of the simultaneous effects of multiple independent variables on a dependent variable, providing insights into the relative contributions and significance of each factor. By exploring this relationship, we can gain valuable insights into the underlying mechanisms and dynamics that govern the perception of temperature in different environmental conditions. Previous studies have primarily focused on analysing the impact of individual meteorological variables on apparent temperature. However, a comprehensive understanding of the combined effects of multiple factors is necessary for accurate modelling and prediction of apparent temperature. Therefore, this study aims to bridge the existing gap by considering various meteorological variables simultaneously, therefore our research question is: What meteorological factors correlate with the apparent temperature?

1.0.3 Dataset

We have found a suitable dataset on Kaggle.com, which contains 96453 entries of weather data. The dataset can be found here. The cvs file is made up of 12 columns, one representing the response variable 'apparent_temp', six describing predictor variables and three which are irrelevant to our research question. The first column represents the date when the data entry was formatted, therefore it will be ignored. 'Loud Cover' is populated only by 0's, therefore it will not be taken into consideration for this data analysis. In the CSV file, besides the relevant numerical variables, there is categorical data as well: "summary", "precip_type" and "daily_summary". The two summaries will be left out, as they just summarise the meteorological data in words. The "precip" variable will be transformed to a dummy variable with three levels, as follows: {'null': 0, 'rain': 1, 'snow': 2}.

[28]:

```
weather = pd.read_csv('weather.csv')

#summarising and cleaning up the dataset
print('The data set contains: ', len(weather['temp']))

#Removing the Formatted, Loud Cover and Summary columns
weather = weather.drop('formatted_date', axis = 1)
weather = weather.drop('loud_cover', axis = 1)
weather = weather.drop('daily_summary', axis = 1)
weather = weather.drop('summary', axis = 1)
```

```
The data set contains:  96453
```

[29]:

```
#converting categorical data to dummy variables



dummy_coding_precip = {'none': 0, 'rain': 1, 'snow': 2}
precip_dummy = weather['precip'].copy()
precip_dummy = precip_dummy.replace(dummy_coding_precip)
weather['precip_dummy'] = precip_dummy

weather.head()
```

[29]:

```
  precip       temp  apparent_temp  humidity  wind_speed  wind_bearing  \
0   none  19.016667      19.016667      0.81     14.8764           163
1   none  17.850000      17.850000      0.81     13.7977           169
```

```
2   none  16.322222       16.322222       0.81      10.8192              151
3   none  12.566667       12.566667       0.81       9.0160              159
4   none  12.927778       12.927778       0.81      17.6295              197

   visibility  air_pressure  precip_dummy
0       9.982       1002.40             0
1       9.982       1001.79             0
2       9.982       1001.60             0
3       9.982       1001.92             0
4      16.100       1002.20             0
```

1.0.4 Explanatory Variables

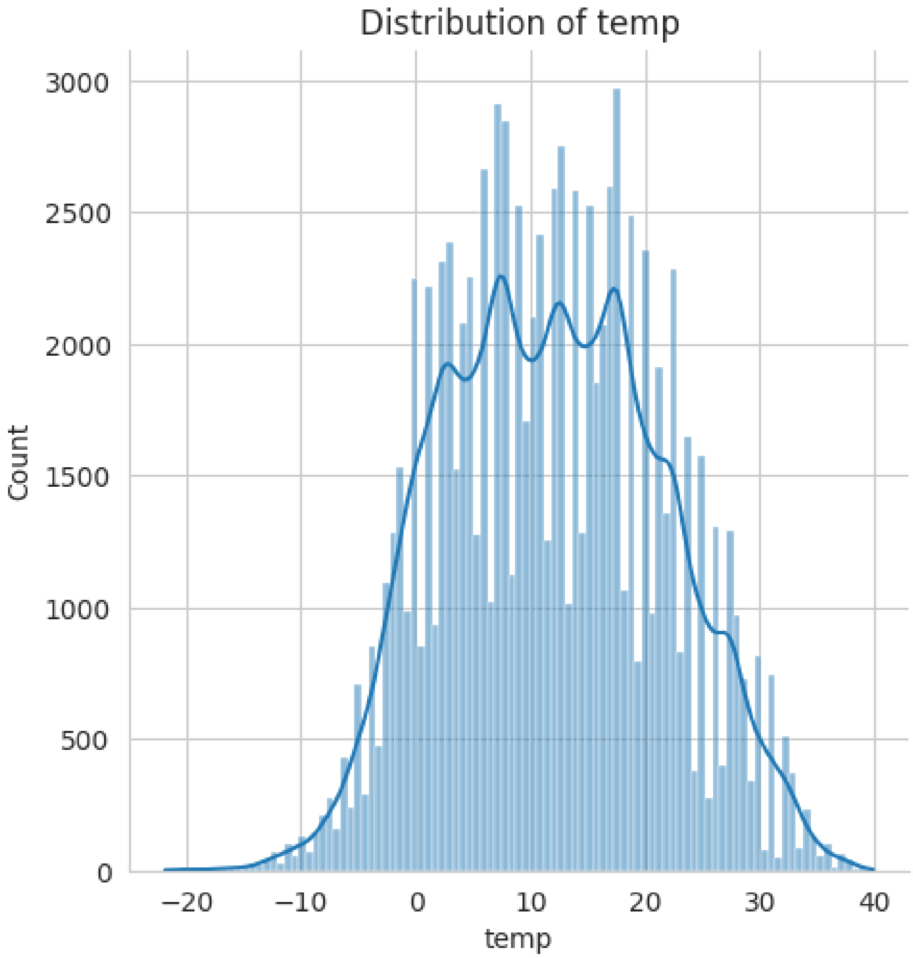
We will use six explanatory variables from our chosen dataset. 'temp' is the actual temperature in Celsius degrees. 'humidity' is the percentage of air humidity. 'wind_speed' is measured in kilometres per hour. 'air_pressure' is the atmospheric pressure, measured in millibars and visibility represents how far someone can see, in kilometres. All the aforementioned variables are continuous. There is also one discrete explanatory variable, the wind bearing, measured in degrees. This indicates the angle at which the wind is blowing.

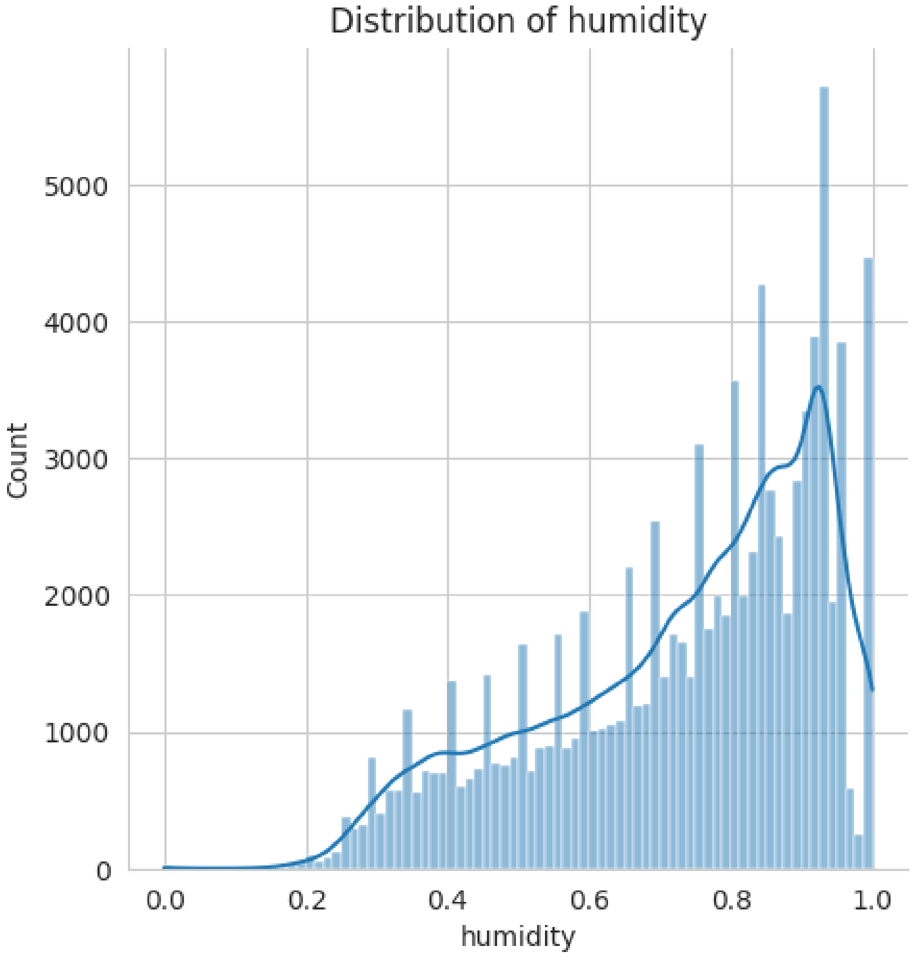
The circumstances under which the data was collected are not disclosed on Kaggle.com, however the data is in line with expectations and consistent with real meteorological conditions.

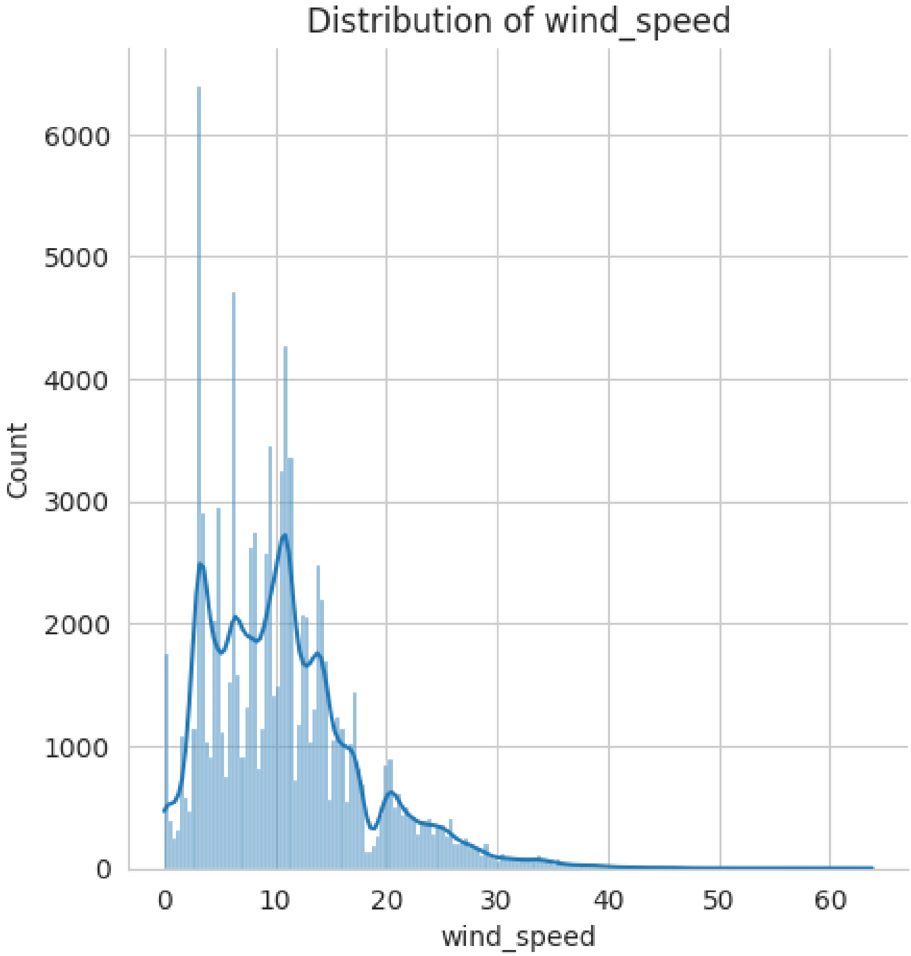
We will perform univariate analysis on the continuous explanatory variables in order to better visualise their distribution.

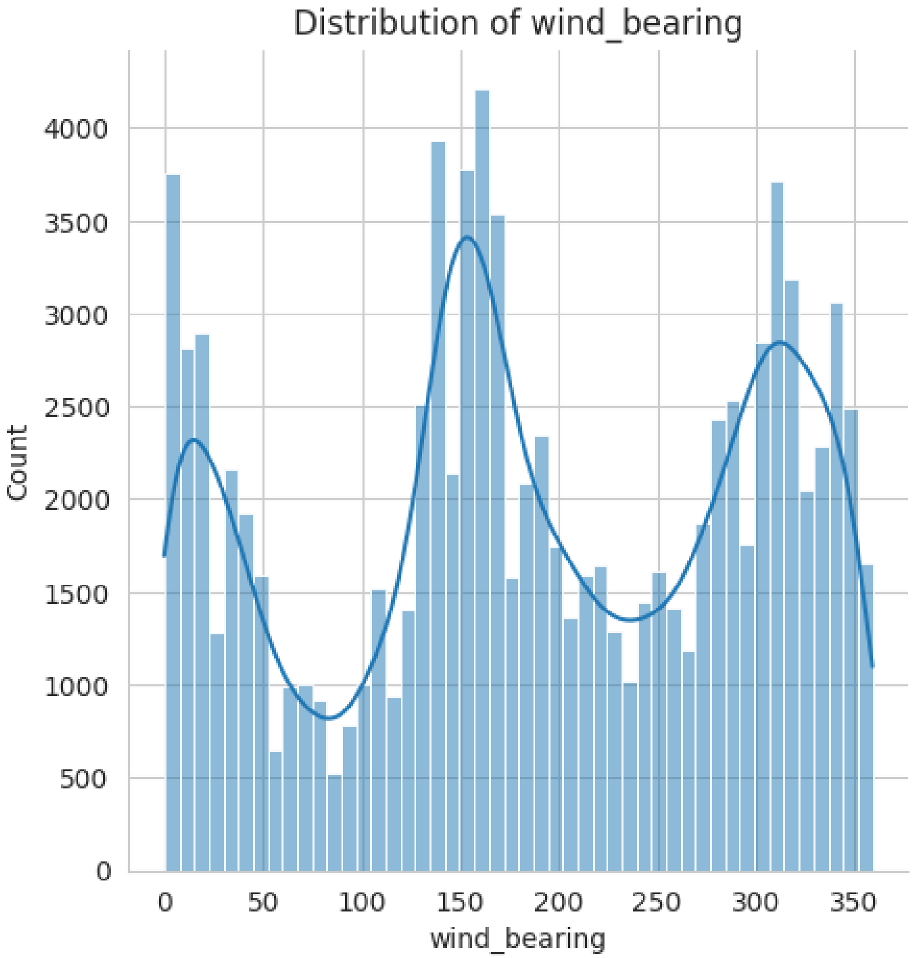
[26]:
```
#performing univariate analysis on each continous explanatory variable

plt.figure(figsize=(4,2))
expl_variables = [ 'temp', 'humidity','wind_speed', 'wind_bearing',␣
 ↪'visibility', 'air_pressure']
for var in expl_variables:
    sns.displot(weather[var], kde = True, rug = False).set(title =␣
 ↪'Distribution of {}'.format(var))
```

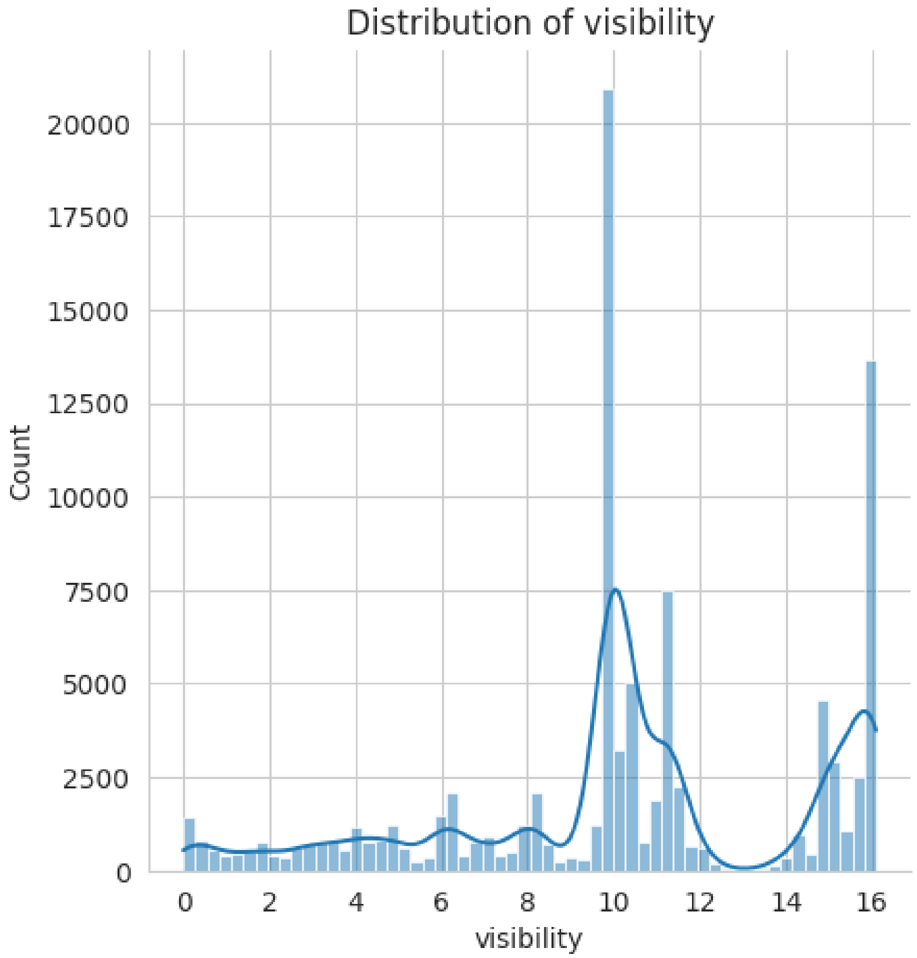
```
<Figure size 400x200 with 0 Axes>
```

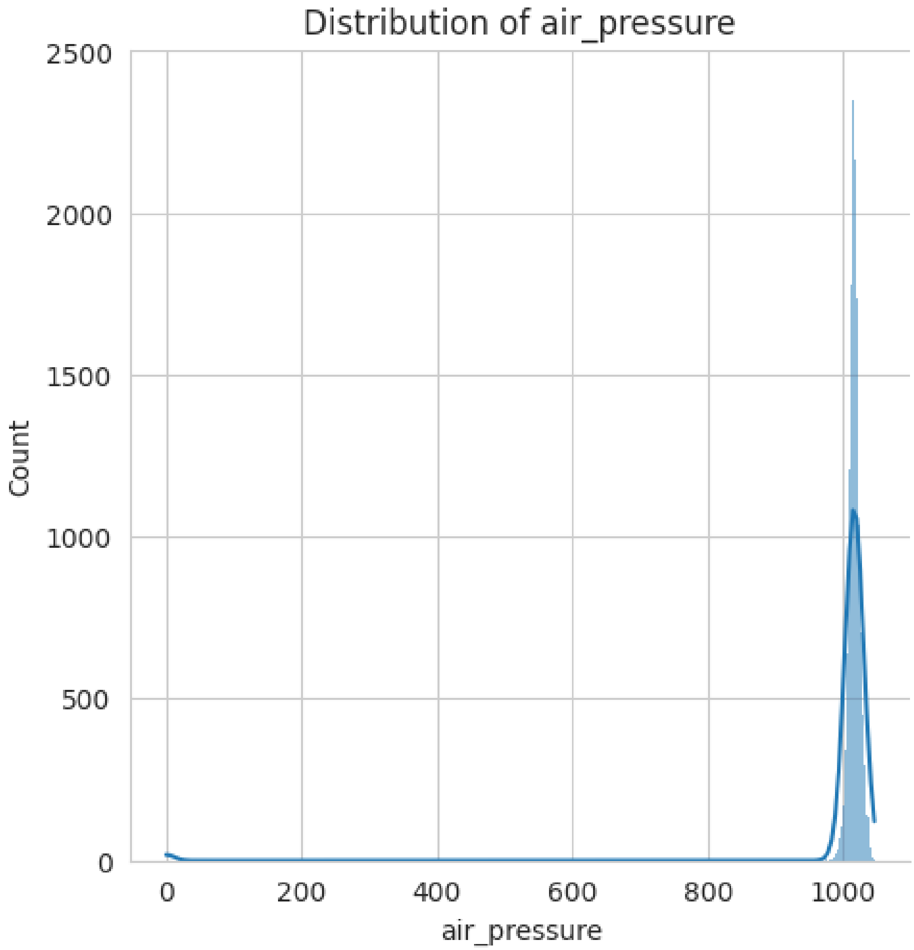

Distribution of temp
Count
3000
2500
2000
1500
1000
500
0
−20
−10
0
10
20
30
40
temp


Distribution of humidity
Count
0
1000
2000
3000
4000
5000
0.0
0.2
0.4
0.6
0.8
1.0
humidity


Distribution of wind_speed
Count
0
1000
2000
3000
4000
5000
6000
0
10
20
30
40
50
60
wind_speed


Distribution of wind_bearing
Count
0
500
1000
1500
2000
2500
3000
3500
4000
0
50
100
150
200
250
300
350
wind_bearing


Distribution of visibility
Count
20000
17500
15000
12500
10000
7500
5000
2500
0
0
2
4
6
8
10
12
14
16
visibility

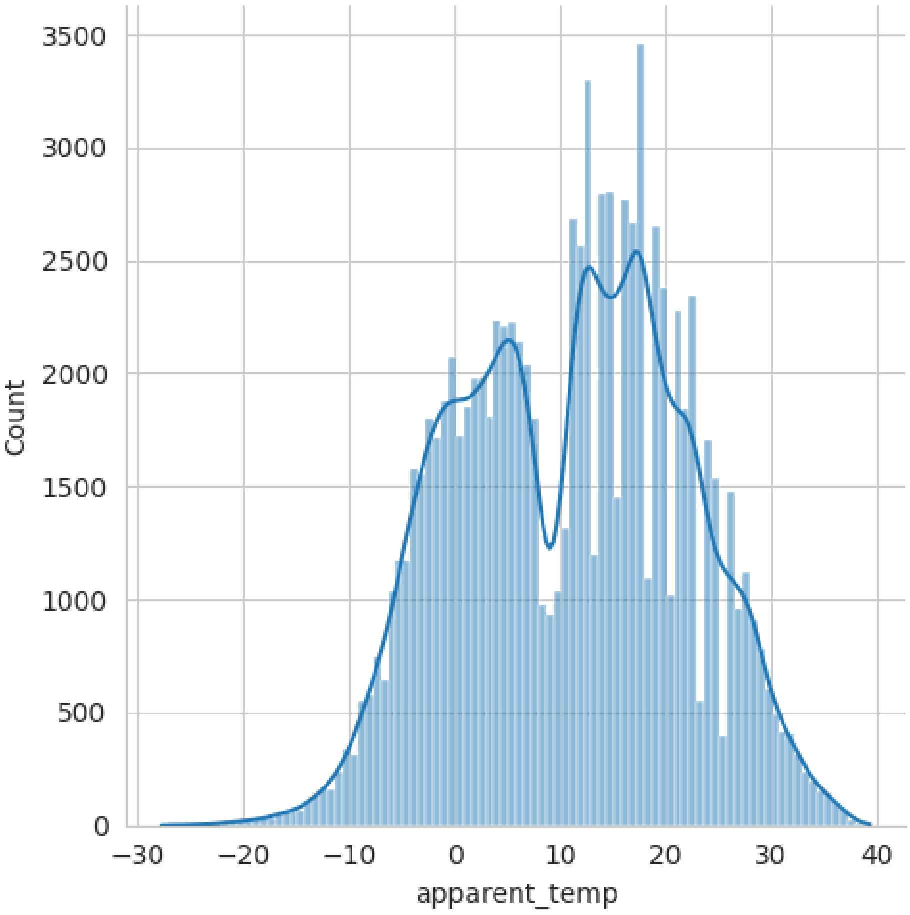


1.0.5 Response Variable

The response variable for which we will study the relation between several factors will be the apparent temperature in Celsius degrees. It is a numerical continuous interval variable. We first plot the variable to check its normality and ensure that no transformation is required.

[30]:
```
#visualising the distribution of the response variable
plt.figure(figsize = (8,5))
sns.displot( x = weather['apparent_temp'], kde = True, rug = False)
plt.show()
```

```
<Figure size 800x500 with 0 Axes>
```



1.0.6 Hypothesis

The hypothesis for each predictor variable is as follows:

Null hypothesis (H0): The coefficient (i) for predictor variable i is equal to zero when considering the other explanatory variables in the model.

Alternative hypothesis (Ha): The coefficient (i) for predictor variable i is not equal to zero when considering the other explanatory variables in the model.

A significance level of =0.05 is chosen to evaluate the significance of the coefficients.

1.0.7 Methods

To answer our research question, we started with the next code cell, where we are implementing a p-value based backward-selection approach to determine the final model for predicting the apparent temperature.

We first fit the full model and observe the initial p-values. We observe that 'humidity' is above the significance level, therefore it is eliminated according to the backward selection algorithm, after which the model is refitted.

```
[31]: #performing backward-selection on the dataset in order to determine the final␣
        ↪model

      #fit the full model

      m_full = sm.formula.ols(formula = 'apparent_temp ~ precip_dummy + temp +␣
        ↪humidity + wind_speed + wind_bearing + visibility + air_pressure ', data =␣
        ↪weather)
      multi_reg = m_full.fit()
      print(multi_reg.summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:          apparent_temp   R-squared:                       0.990
Model:                            OLS   Adj. R-squared:                  0.990
Method:                 Least Squares   F-statistic:                 1.342e+06
Date:                Fri, 16 Jun 2023   Prob (F-statistic):               0.00
Time:                        12:32:06   Log-Likelihood:            -1.4414e+05
No. Observations:               96453   AIC:                         2.883e+05
Df Residuals:                   96445   BIC:                         2.884e+05
Df Model:                           7
Covariance Type:            nonrobust
================================================================================
                   coef    std err          t      P>|t|      [0.025      0.975]
--------------------------------------------------------------------------------
Intercept       -0.8395      0.039    -21.549      0.000      -0.916      -0.763
precip_dummy    -0.5331      0.013    -41.464      0.000      -0.558      -0.508
temp             1.1041      0.000   2433.569      0.000       1.103       1.105
humidity         0.0148      0.018      0.826      0.409      -0.020       0.050
wind_speed      -0.1032      0.001   -202.515      0.000      -0.104      -0.102
wind_bearing     0.0006   3.26e-05     18.238      0.000       0.001       0.001
visibility      -0.0103      0.001    -11.207      0.000      -0.012      -0.008
air_pressure     0.0002   2.98e-05      6.982      0.000       0.000       0.000
==============================================================================
Omnibus:                     2171.378   Durbin-Watson:                   0.443
Prob(Omnibus):                  0.000   Jarque-Bera (JB):             2452.917
Skew:                           0.337   Prob(JB):                         0.00
Kurtosis:                       3.397   Cond. No.                     1.18e+04
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly
specified.
[2] The condition number is large, 1.18e+04. This might indicate that there are
strong multicollinearity or other numerical problems.
```

Subsequently, all the p-values remained are 0.000, indicating that they have high statistical significance. We observe however that the coefficients of 'wind_bearing' and 'air_pressure' are extremely low, indicating that they have little to no impact on the response variable. We examined the option of eliminating them from our model by refitting it three times: twice without each of the variables and once without both of them. Since the results were not influenced a significant amount, we made the decision to indeed eliminate these two variables.

In the end, we obtained the final model which is left with four relevant explanatory variables: the precipitation type, actual temperature, wind speed and visibility ahead. All of these have a p-value of 0, indicating high statistical importance, as well as coefficients which have an impact on the prediction of ‘apparent_temp’.

[32]:

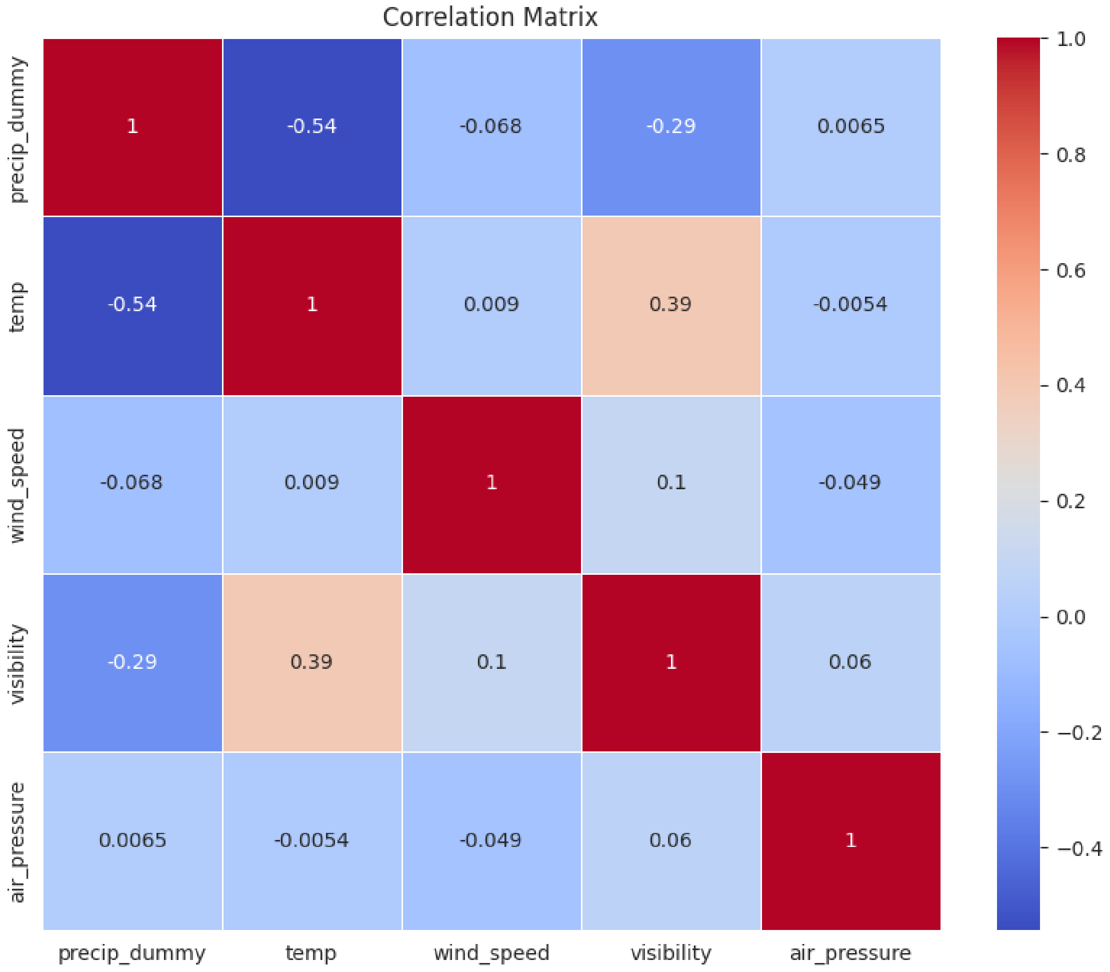
```
#wind_bearing and air_pressure have very low coefficients. we fit the model␣
 ↪without them and see their importance
#we can conclude that their impact is negligeble and exclude them from the␣
 ↪final model, namely m4

m_final = sm.formula.ols(formula =  'apparent_temp ~ precip_dummy + temp +␣
 ↪wind_speed + visibility', data = weather)
m_final_fitted = m_final.fit()
print(m_final_fitted.summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:          apparent_temp   R-squared:                       0.990
Model:                            OLS   Adj. R-squared:                  0.990
Method:                 Least Squares   F-statistic:                 2.339e+06
Date:                Fri, 16 Jun 2023   Prob (F-statistic):               0.00
Time:                        12:32:06   Log-Likelihood:            -1.4433e+05
No. Observations:               96453   AIC:                         2.887e+05
Df Residuals:                   96448   BIC:                         2.887e+05
Df Model:                           4
Covariance Type:            nonrobust
================================================================================
                   coef    std err          t      P>|t|      [0.025      0.975]
--------------------------------------------------------------------------------
Intercept       -0.5208      0.021    -24.920      0.000      -0.562      -0.480
precip_dummy    -0.5374      0.013    -41.792      0.000      -0.563      -0.512
temp             1.1040      0.000   2431.177      0.000       1.103       1.105
wind_speed      -0.1025      0.001   -202.026      0.000      -0.103      -0.101
visibility      -0.0093      0.001    -10.194      0.000      -0.011      -0.008
================================================================================
Omnibus:                     2133.044   Durbin-Watson:                   0.440
Prob(Omnibus):                  0.000   Jarque-Bera (JB):             2404.215
Skew:                           0.334   Prob(JB):                         0.00
Kurtosis:                       3.391   Cond. No.                         144.
==============================================================================


Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly
```

specified.

At first glance, everything seems in order with the resulting model. To ensure that there is no high correlation between our variables. We examined their relationship using a correlation matrix as a heatmap, which can be seen below. The low intensity of the colours indicates that there is indeed no high correlation, which further confirms that our model is appropriately constructed.

[33]:
```python
#bivariate analysis between explanatory variables

# Compute correlation coefficients
corr_matrix = weather[['precip_dummy', 'temp', 'wind_speed', 'visibility',␣
 ↪'air_pressure']].corr()

# Plot correlation matrix as a heatmap
plt.figure(figsize=(10, 8))
sns.heatmap(corr_matrix, annot=True, cmap='coolwarm', linewidths=0.5)
plt.title('Correlation Matrix')
plt.show()

#colder colours show that none are highly correlated
```



1.0.8 Residuals Analysis

Before we go ahead with reporting the results of our analysis, we must first ensure that the following assumptions regarding the residuals in the model are met:

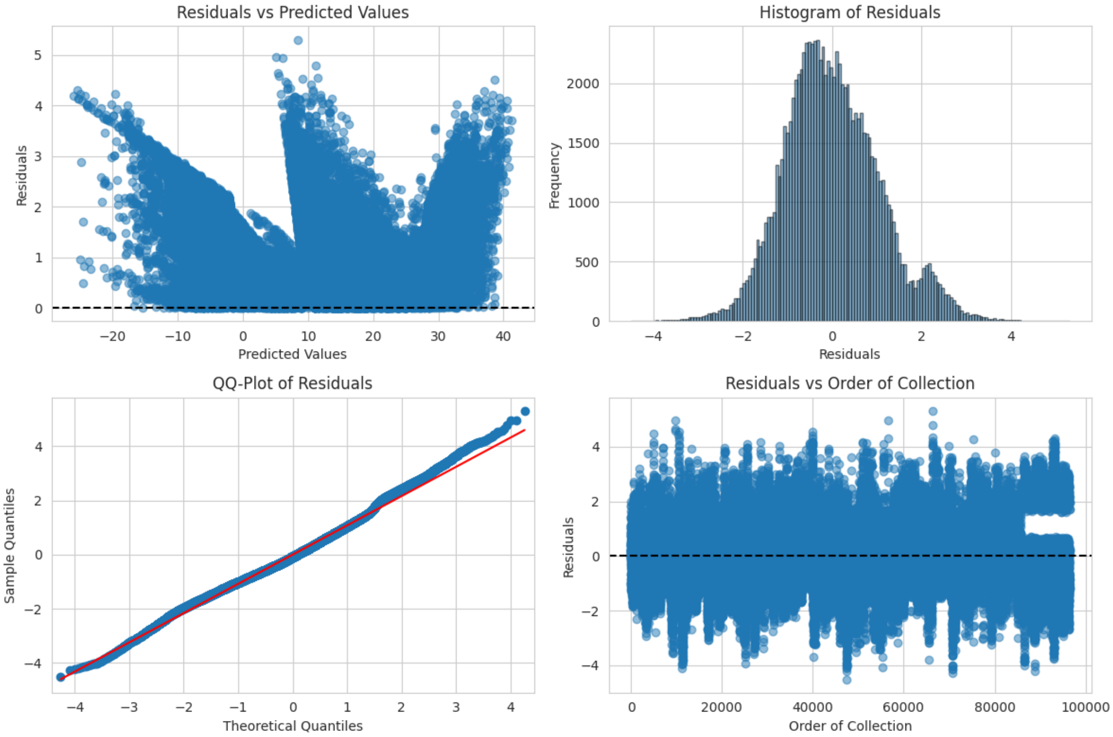
- Variability of the residuals is nearly constant
- The distribution of the residuals is nearly normal
- Residuals are independent
- Each variable is linearly related to the outcome

In order to check the variability, we will generate a scatterplot of the absolute residuals against the predicted values in the model. We will ensure they are distributed normally using a QQ-plot and a histogram. Their independence can be verified by creating a scatterplot where the residuals are presented in the order of collection.

[34]:

```
#before reporting model results, we check that the model conditions are verrified

residuals = m_final_fitted.resid
fig, axes = plt.subplots(2, 2, figsize=(12, 8))
#constant variability of residuals
axes[0, 0].scatter(m_final_fitted.fittedvalues, np.abs(residuals), alpha=0.5)
axes[0, 0].axhline(y=0, color='black', linestyle='--')
axes[0, 0].set_xlabel('Predicted Values')
axes[0, 0].set_ylabel('Residuals')
axes[0, 0].set_title('Residuals vs Predicted Values')

#normally distributed residuals
axes[0, 1].hist(residuals, bins='auto', alpha=0.5, edgecolor='black')
axes[0, 1].set_xlabel('Residuals')
axes[0, 1].set_ylabel('Frequency')
axes[0, 1].set_title('Histogram of Residuals')

sm.qqplot(residuals, line='s', ax=axes[1, 0])
axes[1, 0].set_xlabel('Theoretical Quantiles')
axes[1, 0].set_ylabel('Sample Quantiles')
axes[1, 0].set_title('QQ-Plot of Residuals')

#independent residuals
axes[1, 1].scatter(range(len(residuals)), residuals, alpha=0.5)
axes[1, 1].axhline(y=0, color='black', linestyle='--')
axes[1, 1].set_xlabel('Order of Collection')
axes[1, 1].set_ylabel('Residuals')
axes[1, 1].set_title('Residuals vs Order of Collection')
```

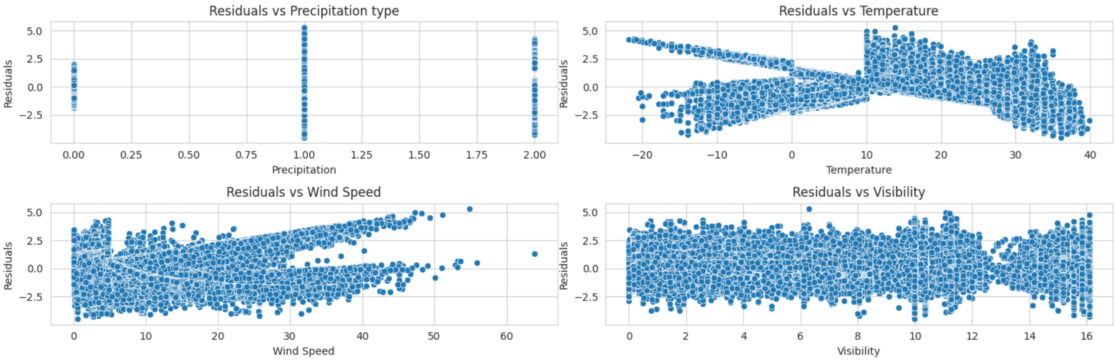
```
plt.tight_layout()
```



Finally, to check the linear relationship between the residuals and the outcome, we will plot them against each explanatory variable using scatterplots.

[35]:
```
# Each variable is linearly related to the outcome
fig, axes = plt.subplots(2, 2, figsize=(15, 5))

#precip_dummy
sns.scatterplot(x=weather['precip_dummy'], y=residuals, ax=axes[0][0])
axes[0][0].set_xlabel('Precipitation')
axes[0][0].set_ylabel('Residuals')
axes[0][0].set_title('Residuals vs Precipitation type')
# temp
sns.scatterplot(x=weather['temp'], y=residuals, ax=axes[0][1])
axes[0][1].set_xlabel('Temperature')
axes[0][1].set_ylabel('Residuals')
axes[0][1].set_title('Residuals vs Temperature')

# wind_speed
sns.scatterplot(x=weather['wind_speed'], y=residuals, ax=axes[1][0])
axes[1][0].set_xlabel('Wind Speed')
```

```
axes[1][0].set_ylabel('Residuals')
axes[1][0].set_title('Residuals vs Wind Speed')

# visibility
sns.scatterplot(x=weather['visibility'], y=residuals, ax=axes[1][1])
axes[1][1].set_xlabel('Visibility')
axes[1][1].set_ylabel('Residuals')
axes[1][1].set_title('Residuals vs Visibility')

plt.tight_layout()
plt.show()
```



It can be observed that while the graphs are not fully in line with our assumptions, they are reasonably so, which allows us to go forward with reporting the final results of our model. Concerns regarding the distribution of the explanatory variables and residuals will be further addressed in the 'Limitations' section of our report.

1.0.9 Results

The final linear equation resulted from our model is as follows:

The R-squared value of our model is 0.99, which indicates that there is a strong correlation between the selected explanatory variables and the response variable.

The interpretation of the intercept makes sense in the context of our model. In the case where all explanatory variables are 0, the apparent temperature is predicted to be -0.48 degrees Celsius.

The slope associated with the dummy variable for precipitations indicates that, all other variables held constant, rainy days are predicted to have a temperature lower by 0.56 degrees, meanwhile on snowy days the apparent temperature decreases by double that, 1.12 degrees Celsius. For each one degree increase in the real temperature, the apparent is projected to increase by 1.1 . Wind speed and visibility are expected to decrease the real-feel temperature by 0.1 and 0.01 respectively, with each unit increase.

1.0.10 Conclusion

Our research set out to answer the question of what meteorological factors could be used to accurately predict the apparent, also called real-feel, temperature on a given day. Starting from a full model, we used the backwards-selection approach to eliminate the factors which possessed no significant association to our response variable and reach our final regression model. According to our model, the apparent temperature is influenced by the actual measured temperature, the precipitation type, wind speed and visibility. Of all the predictor variables considered, the measure temperature was shown to be the most influential factor in our prediction. This aligns with what we expected, as temperature is what mainly determines how temperature feels to humans.

1.0.11 Limitations

One significant concern is the potential presence of biased data. The source of the dataset does not provide information regarding when and where the measurements were taken, which introduces potential biases. The lack of specificity regarding timing and location may result in unaccounted-for variations in the data, impacting the accuracy and generalizability of the model.

Another limitation is the problem of the distribution of residuals and their apparently non-linear relation with the explanatory variables. Although a multi-linear model or logistic regression could potentially address this problem, such approaches were not used due to being outside the scope of the course and the results obtained still being reasonable. Consequently, the regression model may not accurately capture the relationships between the predictors and the response variable, potentially leading to less accurate predictions. Moreover, it is possible that there exist other variables which have an influence on our response variable but were not included in the dataset, fact which can potentially have a negative impact on the accuracy of our predictions.

Another important limitation regards the generalizability of our results. Due to lack of specific information about the context in which the data was collected and the sampling procedure, it is difficult to determine to what extent the findings can be applied to different populations, locations, or time periods.