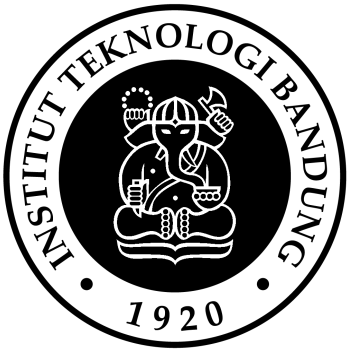
IF2220 Probabilitas dan Statistika

ANALISIS DESKRIPTIF, PENARIKAN KESIMPULAN, DAN PENGUJIAN HIPOTESIS PADA DATASET MINUMAN ANGGUR

Laporan Tugas Besar

Disusun untuk memenuhi tugas mata kuliah Probabilitas dan Statistika
pada Semester II (dua) Tahun Akademik 2022/2023.



Oleh

Alisha Listya Wardhani	13521171
Chiquita Ahsanunnisa	13521129

PROGRAM STUDI TEKNIK INFORMATIKA
SEKOLAH TEKNIK ELEKTRO DAN INFORMATIKA
INSTITUT TEKNOLOGI BANDUNG
BANDUNG

2023

1

April 18, 2023

[1]:
```
import pandas as pd

dataAnggur = pd.read_csv('../data/anggur.csv')
```

Menulis deskripsi statistika (Descriptive Statistics) dari semua kolom pada data yang bersifat numerik, terdiri dari mean, median, modus, standar deviasi, variansi, range, nilai minimum, maksimum, kuartil, IQR, skewness dan kurtosis. Boleh juga ditambahkan deskripsi lain.

0.0.1 1. Mean

[2]:
```
dataAnggur.mean()
```

[2]:
```
fixed acidity            7.152530
volatile acidity         0.520839
citric acid              0.270517
residual sugar           2.567104
chlorides                0.081195
free sulfur dioxide     14.907679
total sulfur dioxide    40.290150
density                  0.995925
pH                       3.303610
sulphates                0.598390
alcohol                 10.592280
quality                  7.958000
dtype: float64
```

0.0.2 2. Median

[3]:
```
dataAnggur.median()
```

[3]:
```
fixed acidity            7.150000
volatile acidity         0.524850
citric acid              0.272200
residual sugar           2.519430
chlorides                0.082167
free sulfur dioxide     14.860346
total sulfur dioxide    40.190000
density                  0.996000
```

```
pH                      3.300000
sulphates               0.595000
alcohol                10.610000
quality                 8.000000
dtype: float64
```

0.0.3 3. Modus

[4]:
```
dataAnggur.mode().iloc[0]
```

[4]:
```
fixed acidity           6.540000
volatile acidity        0.554600
citric acid             0.301900
residual sugar          0.032555
chlorides               0.015122
free sulfur dioxide     0.194679
total sulfur dioxide   35.200000
density                 0.995900
pH                      3.340000
sulphates               0.590000
alcohol                 9.860000
quality                 8.000000
Name: 0, dtype: float64
```

0.0.4 4. Standar Deviasi

[5]:
```
dataAnggur.var()
```

[5]:
```
fixed acidity           1.443837
volatile acidity        0.009187
citric acid             0.002411
residual sugar          0.975977
chlorides               0.000404
free sulfur dioxide    23.893519
total sulfur dioxide   99.316519
density                 0.000004
pH                      0.010999
sulphates               0.010164
alcohol                 2.282233
quality                 0.815051
dtype: float64
```

0.0.5 5. Variansi

[6]:
```
dataAnggur.std()
```

[6]:
```
fixed acidity           1.201598
volatile acidity        0.095848
citric acid             0.049098
residual sugar          0.987915
chlorides               0.020111
free sulfur dioxide     4.888100
total sulfur dioxide    9.965767
density                 0.002020
pH                      0.104875
sulphates               0.100819
alcohol                 1.510706
quality                 0.902802
dtype: float64
```

0.0.6 6. Range

[7]:
```
dataAnggur.max() - dataAnggur.min()
```

[7]:
```
fixed acidity           8.170000
volatile acidity        0.665200
citric acid             0.292900
residual sugar          5.518200
chlorides               0.125635
free sulfur dioxide    27.267847
total sulfur dioxide   66.810000
density                 0.013800
pH                      0.740000
sulphates               0.670000
alcohol                 8.990000
quality                 5.000000
dtype: float64
```

0.0.7 7. Nilai Minimum

[8]:
```
dataAnggur.min()
```

[8]:
```
fixed acidity           3.320000
volatile acidity        0.139900
citric acid             0.116700
residual sugar          0.032555
chlorides               0.015122
free sulfur dioxide     0.194679
total sulfur dioxide    3.150000
density                 0.988800
pH                      2.970000
sulphates               0.290000
alcohol                 6.030000
```

```
quality                 5.000000
dtype: float64
```

0.0.8 8. Nilai Maksimum

[9]:
```
dataAnggur.max()
```

[9]:
```
fixed acidity           11.490000
volatile acidity         0.805100
citric acid              0.409600
residual sugar           5.550755
chlorides                0.140758
free sulfur dioxide     27.462525
total sulfur dioxide    69.960000
density                  1.002600
pH                       3.710000
sulphates                0.960000
alcohol                 15.020000
quality                 10.000000
dtype: float64
```

0.0.9 9. Quartil

Pada tabel di bawah ini, 0.25 bermakna quartil pertama (25%), 0.50 bermakna quartil kedua (50%), dan 0.75 bermakna quartil ketiga (75%).

[10]:
```
dataAnggur.quantile([0.25,0.50,0.75])
```

[10]:
```
      fixed acidity  volatile acidity  citric acid  residual sugar  chlorides  \
0.25         6.3775          0.456100     0.237800        1.896330   0.066574
0.50         7.1500          0.524850     0.272200        2.519430   0.082167
0.75         8.0000          0.585375     0.302325        3.220873   0.095312

      free sulfur dioxide  total sulfur dioxide  density    pH  sulphates  \
0.25            11.426717               33.7850   0.9946  3.23      0.530
0.50            14.860346               40.1900   0.9960  3.30      0.595
0.75            18.313098               47.0225   0.9972  3.37      0.670

      alcohol  quality
0.25   9.5600      7.0
0.50  10.6100      8.0
0.75  11.6225      9.0
```

0.0.10 10. Inter Quartile Range (IQR)

[11]:
```
dataAnggur.quantile(0.75) - dataAnggur.quantile(0.25)
```

[11]:
```
fixed acidity           1.622500
volatile acidity        0.129275
citric acid             0.064525
residual sugar          1.324544
chlorides               0.028738
free sulfur dioxide     6.886381
total sulfur dioxide   13.237500
density                 0.002600
pH                      0.140000
sulphates               0.140000
alcohol                 2.062500
quality                 2.000000
dtype: float64
```

0.0.11 11. Skewness

[12]:
```
dataAnggur.skew()
```

[12]:
```
fixed acidity          -0.028879
volatile acidity       -0.197699
citric acid            -0.045576
residual sugar          0.132638
chlorides              -0.051319
free sulfur dioxide     0.007130
total sulfur dioxide   -0.024060
density                -0.076883
pH                      0.147673
sulphates               0.149199
alcohol                -0.018991
quality                -0.089054
dtype: float64
```

0.0.12 12. Excess Kurtosis

[13]:
```
dataAnggur.kurtosis()
```

[13]:
```
fixed acidity          -0.019292
volatile acidity        0.161853
citric acid            -0.104679
residual sugar         -0.042980
chlorides              -0.246508
free sulfur dioxide    -0.364964
total sulfur dioxide    0.063950
density                 0.016366
pH                      0.080910
sulphates               0.064819
alcohol                -0.131732
```

```
quality                 0.108291
dtype: float64
```

2

April 18, 2023

[1]:
```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

dataAnggur = pd.read_csv('../data/anggur.csv')
```

Membuat Visualisasi plot distribusi, dalam bentuk histogram dan boxplot untuk setiap kolom numerik. Berikan uraian penjelasan kondisi setiap kolom berdasarkan kedua plot tersebut.

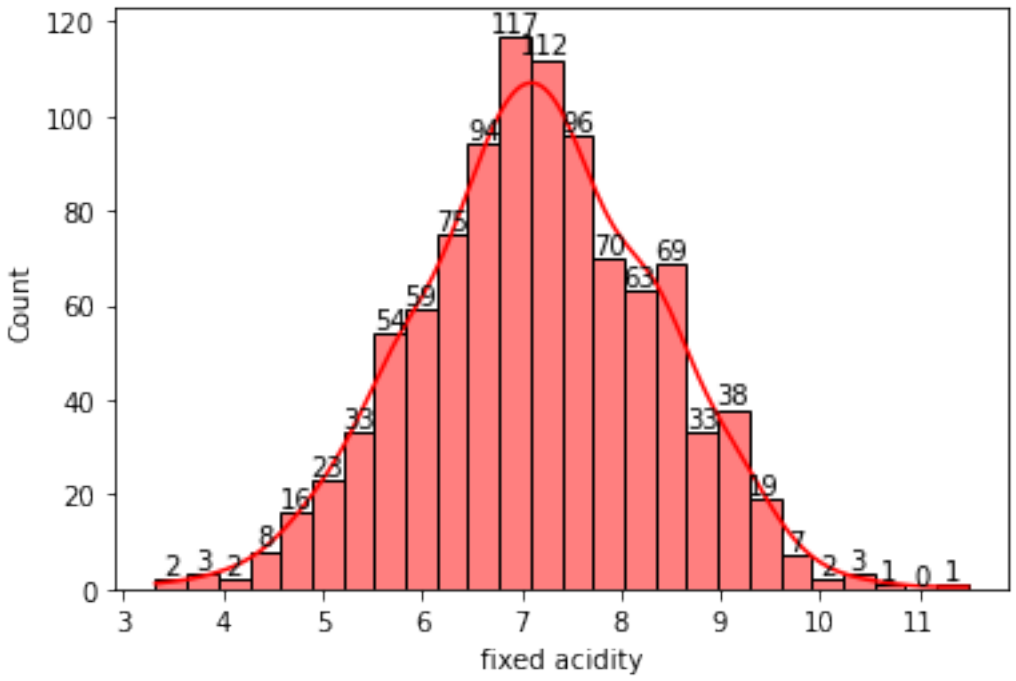
0.0.1 1. fixed acidity

[2]:
```
# Data
dataFixedAcidity = dataAnggur['fixed acidity']
```

[3]:
```
# ======= Histogram ============
ax = sns.histplot(dataFixedAcidity, color='red', stat = 'count', kde = True)
for i in ax.containers:
    ax.bar_label(i,)

# Print the bin edges
bin_edges = [patch.get_x() for patch in ax.patches]
print("Bin Edges (from leftmost): ", bin_edges)
```

```
Bin Edges (from leftmost):  [3.32, 3.6342307692307685, 3.948461538461538,
4.2626923076923084, 4.576923076923077, 4.891153846153847, 5.205384615384615,
5.519615384615385, 5.833846153846153, 6.148076923076923, 6.462307692307691,
6.776538461538461, 7.090769230769231, 7.404999999999999, 7.719230769230769,
8.033461538461538, 8.347692307692308, 8.661923076923078, 8.976153846153846,
9.290384615384614, 9.604615384615386, 9.918846153846154, 10.233076923076922,
10.547307692307692, 10.861538461538462, 11.17576923076923]
```

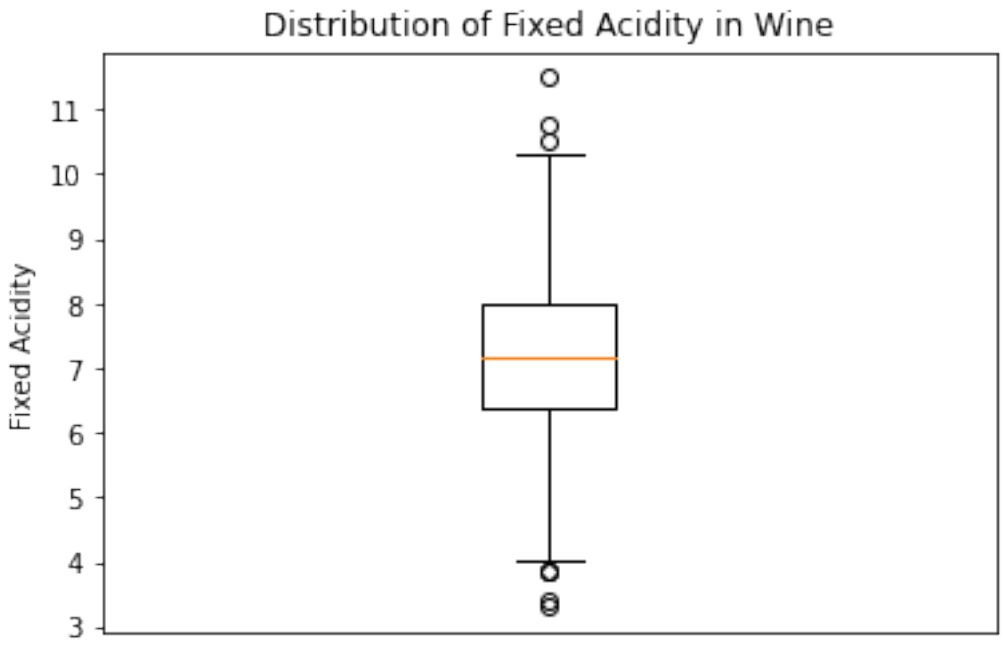


Histogram menunjukkan distribusi nilai *fixed acidity* atau nilai keasaman tetap dalam 1000 sampel anggur. Distribusi tersebut memiliki bentuk *bell-shaped* yang simetris dan memiliki nilai puncak pada 6.77-7.09 (dengan frekuensi 117). Rentang nilai berkisar antara 3,32 hingga 11,49. Ada beberapa yang memiliki nilai *fixed acidity* yang sangat rendah ataupun sangat tinggi tetapi tidak mempengaruhi bentuk keseluruhan distribusi.

[4]:
```
# ======= Boxplot ============
plt.boxplot(dataFixedAcidity)

# Set attributes
plt.title('Distribution of Fixed Acidity in Wine')
plt.ylabel('Fixed Acidity')
plt.xticks([],[])

# Show graph
plt.show()
```



Boxplot menunjukkan nilai minimum dari *fixed acidity* adalah sekitar 4, sedangkan nilai maksimumnya sekitar 10,5. Walaupun begitu, terdapat lima *outlier*, tiga diantaranya berada dibawah nilai minimum. Nilai median terletak pada sekitar 7, dengan *interquartile range* sebesar 6,5 sampai 8.

0.0.2 2. volatile activity

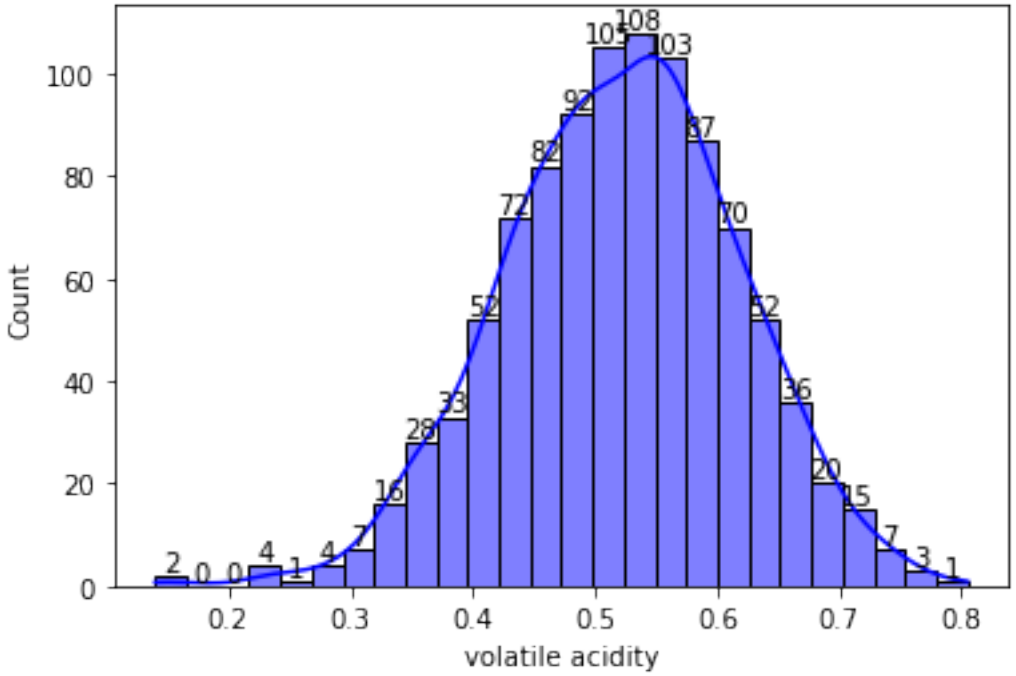
[5]:
```
# Data
dataVolatileAcidity = dataAnggur['volatile acidity']
```

[6]:
```
# ======= Histogram ============
ax = sns.histplot(dataVolatileAcidity, color='blue', stat = 'count', kde = True)
for i in ax.containers:
    ax.bar_label(i,)

# Print the bin edges
bin_edges = [patch.get_x() for patch in ax.patches]
print("Bin Edges (from leftmost): ", bin_edges)
```

```
Bin Edges (from leftmost):  [0.13989999999999997, 0.1654846153846154,
0.19106923076923077, 0.21665384615384614, 0.2422384615384615,
0.26782307692307694, 0.29340769230769237, 0.3189923076923077,
0.3445769230769231, 0.37016153846153843, 0.39574615384615386,
0.4213307692307693, 0.4469153846153846, 0.47250000000000003,
0.49808461538461546, 0.5236692307692308, 0.5492538461538462, 0.5748384615384614,
0.600423076923077, 0.6260076923076924, 0.6515923076923076, 0.6771769230769231,
```

```
0.7027615384615385, 0.7283461538461538, 0.7539307692307693, 0.7795153846153847]
```

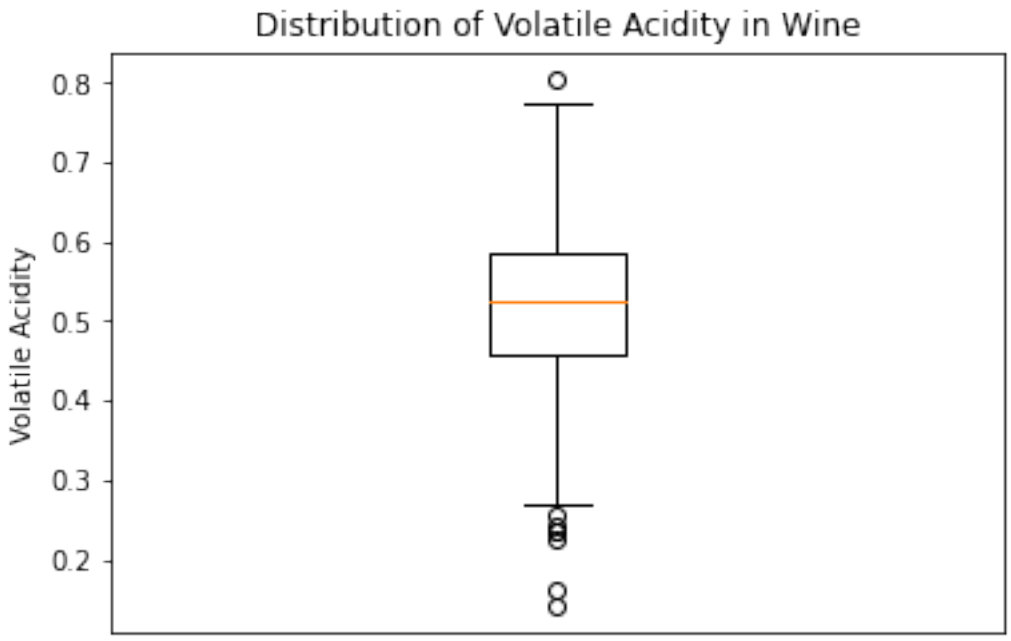


Histogram menunjukkan distribusi nilai *Volatile Acidity* dalam sampel 1000 anggur. Distribusi tersebut terlihat berbentuk *bell-shaped*, dengan distribusi normal. Walaupun begitu, jika dibandingkan dengan histogram kolom *fixed acidity*, bentuk ini sekilas terlihat lebih *negatively skewed*. Distribusi ini memiliki nilai puncak pada range keasaman 0.523 - 0.549, dengan frekuensi 108. Nilai *Volatile Acidity* berkisar antara 0.13 - 0.77.

[7]:
```
# ======= Boxplot ============
plt.boxplot(dataVolatileAcidity)

# Set attributes
plt.title('Distribution of Volatile Acidity in Wine')
plt.ylabel('Volatile Acidity')
plt.xticks([],[])

# Show graph
plt.show()
```



Berdasarkan visualisasi di atas, boxplot menunjukkan nilai minimum distribusi adalah sektar 0.27, sedangkan nilai maksimum terdapat pada 0.78. Nilai median *volatile acidity* berada pada 0.55, dengan *Interquartile Range* sebesar 0.48 - 0.55. Terdapat beberapa *outlier* pada distribusi, kebanyakan memiliki nilai dibawah minimum.

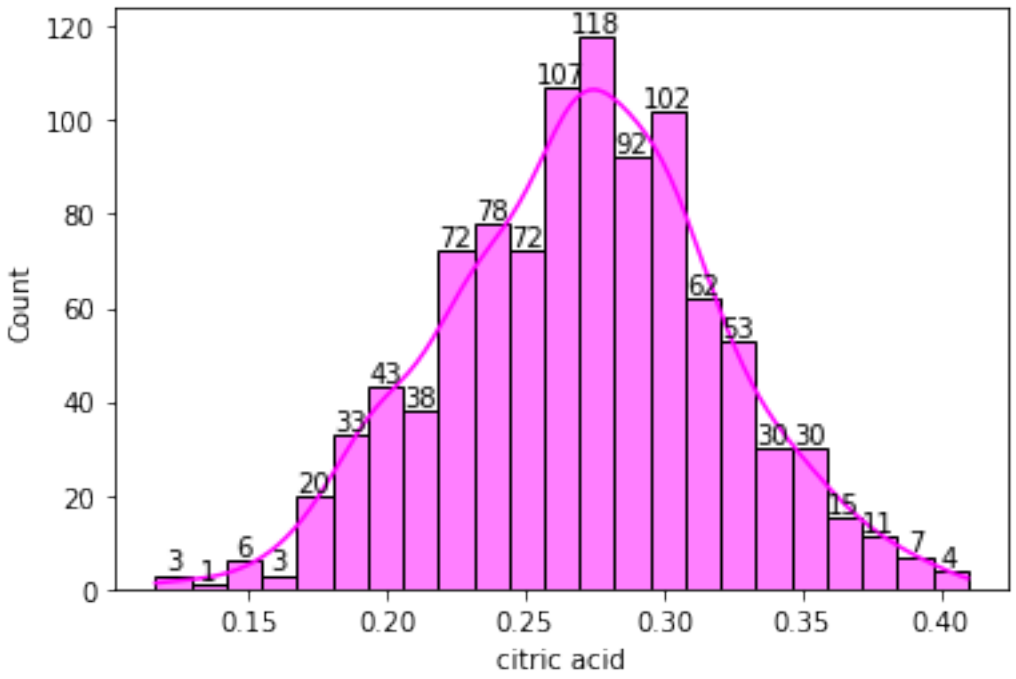
0.0.3 3. citric acid

[8]:
```
# Data
dataCitricAcid = dataAnggur['citric acid']
```

[9]:
```
# ======= Histogram ============
ax = sns.histplot(dataCitricAcid, color='magenta', stat = 'count', kde = True)
for i in ax.containers:
    ax.bar_label(i,)

# Print the bin edges
bin_edges = [patch.get_x() for patch in ax.patches]
print("Bin Edges (from leftmost): ", bin_edges)
```

```
Bin Edges (from leftmost):  [0.1167, 0.12943478260869565, 0.14216956521739132,
0.15490434782608697, 0.16763913043478262, 0.18037391304347827,
0.19310869565217392, 0.20584347826086957, 0.21857826086956522,
0.23131304347826087, 0.24404782608695652, 0.25678260869565217,
0.2695173913043478, 0.28225217391304347, 0.2949869565217392, 0.3077217391304348,
0.3204565217391305, 0.3331913043478261, 0.3459260869565218, 0.3586608695652175,
0.3713956521739131, 0.3841304347826088, 0.396865217391304]
```

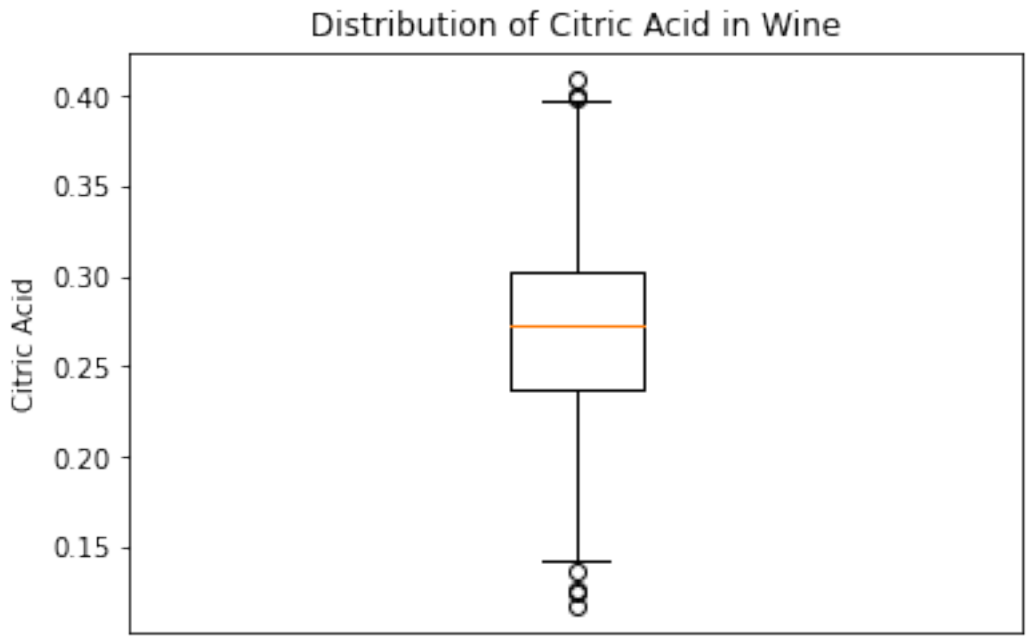


Histogram menunjukkan bahwa distribusi nilai *citric acid* dalam 1000 sampel anggur berbentuk *bell-shaped* atau memiliki distribusi normal. Distribusi tersebut mencapai nilai puncak pada tingkat keasaman 0.269 - 0.282 (dengan frekuensi sebanyak 118). Nilai *citric acid* memiliki range sekitar 0.11 - 0.39. Distribusi ini memiliki beberapa nilai dengan tingkat sangat tinggi ataupun sangat rendah, namun tidak mempengaruhi bentuk distribusi.

[10]:
```
# ======= Boxplot ============
plt.boxplot(dataCitricAcid)

# Set attributes
plt.title('Distribution of Citric Acid in Wine')
plt.ylabel('Citric Acid')
plt.xticks([],[])

# Show graph
plt.show()
```



Berdasarkan visualisasi di atas, boxplot menunjukkan nilai minimum di sekitar 0.13 dan nilai maksimum di sekitar 0.40. Nilai median berada pada 0.27, dengan *interquartile range* sebesar 0.23 - 0.30. Terdapat beberapa outlier pada distribusi, sebagian besar memiliki nilai lebih kecil daripada nilai minimum.

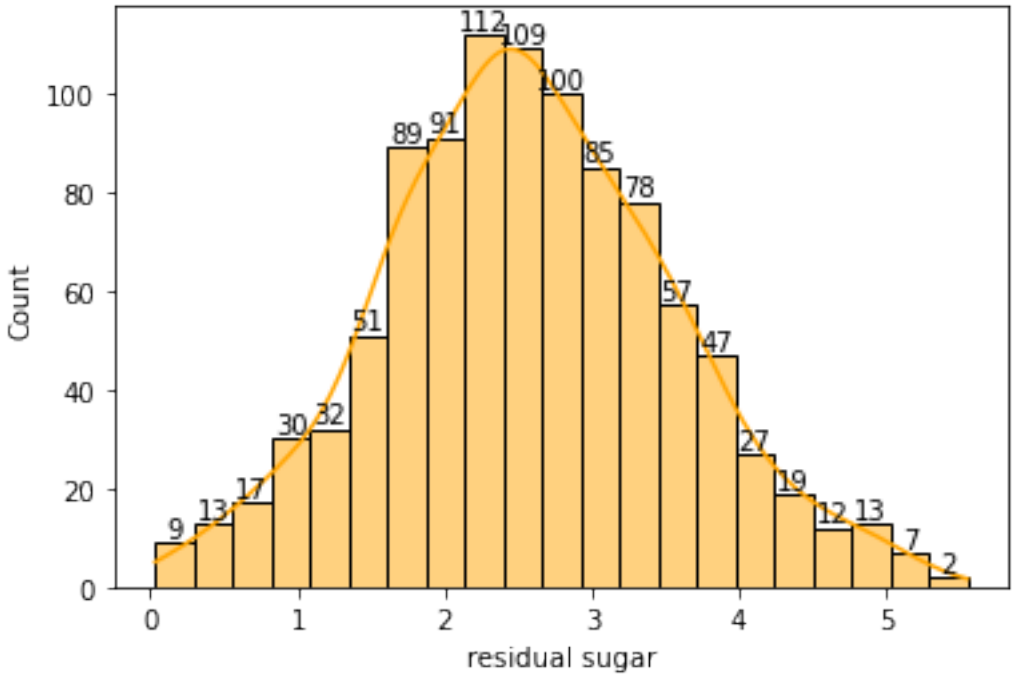
0.0.4 4. residual sugar

[11]:
```
# Data
dataResidualSugar = dataAnggur['residual sugar']
```

[12]:
```
# ======= Histogram ============
ax = sns.histplot(dataResidualSugar, color='orange', stat = 'count', kde = True)
for i in ax.containers:
    ax.bar_label(i,)

# Print the bin edges
bin_edges = [patch.get_x() for patch in ax.patches]
print("Bin Edges (from leftmost): ", bin_edges)
```

```
Bin Edges (from leftmost):  [0.03255452501519501, 0.29532597309652175,
0.5580974211778486, 0.8208688692591752, 1.0836403173405023, 1.3464117654218293,
1.6091832135031559, 1.8719546615844829, 2.1347261096658094, 2.397497557747136,
2.6602690058284626, 2.92304045390979, 3.1858119019911166, 3.448583350072443,
3.71135479815377, 3.974126246235097, 4.236897694316424, 4.4996691423977495,
4.762440590479077, 5.025212038560404, 5.28798348664173]
```

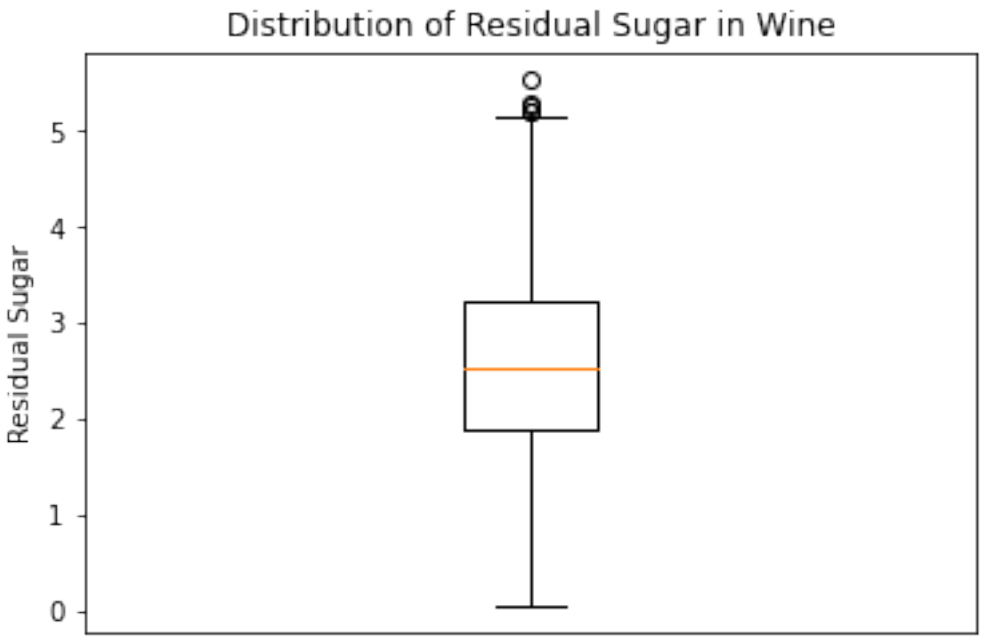


Histogram menunjukkan bahwa distribusi nilai *residual sugar* dalam 1000 sampel anggur berbentuk *bell-shaped* atau memiliki distribusi normal. Distribusi tersebut mencapai nilai puncak pada tingkat residu 2.134 - 2.397 (dengan frekuensi sebanyak 112). Nilai *residual sugar* memiliki range sekitar 0.03 - 5.287. Distribusi ini memiliki beberapa nilai dengan tingkat sangat tinggi ataupun sangat rendah, namun tidak mempengaruhi bentuk distribusi.

[13]:
```
# ======= Boxplot ============
plt.boxplot(dataResidualSugar)

# Set attributes
plt.title('Distribution of Residual Sugar in Wine')
plt.ylabel('Residual Sugar')
plt.xticks([],[])

# Show graph
plt.show()
```



Berdasarkan visualisasi di atas, boxplot menunjukkan nilai minimum distribusi adalah sektar 0, sedangkan nilai maksimum terdapat pada 5.2. Nilai median *residual sugar* berada pada 2.5, dengan *Interquartile Range* diantara 1.8 - 3.2. Terdapat beberapa *outlier* pada distribusi, semua memiliki nilai di atas maksimum.

0.0.5 5. chlorides

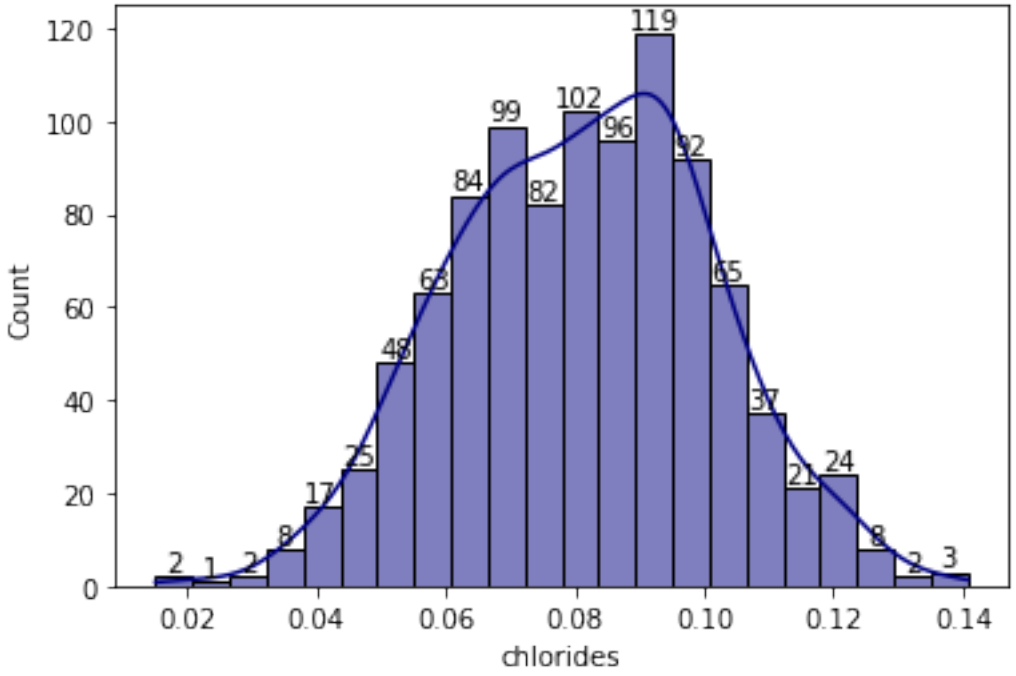
[14]:
```
# Data
dataChlorides = dataAnggur['chlorides']
```

[15]:
```
# ======= Histogram ============
ax = sns.histplot(dataChlorides, color='navy', stat = 'count', kde = True)
for i in ax.containers:
    ax.bar_label(i,)

# Print the bin edges
bin_edges = [patch.get_x() for patch in ax.patches]
print("Bin Edges (from leftmost): ", bin_edges)
```

```
Bin Edges (from leftmost):  [0.0151224391657095, 0.02083312690504354,
0.02654381464437757, 0.03225450238371161, 0.03796519012304565,
0.04367587786237968, 0.049386565601713714, 0.05509725334104776,
0.06080794108038178, 0.06651862881971583, 0.07222931655904988,
0.07794000429838391, 0.08365069203771794, 0.08936137977705198,
0.09507206751638603, 0.10078275525572006, 0.10649344299505409,
0.11220413073438812, 0.11791481847372216, 0.12362550621305621,
```

```
0.12933619395239024, 0.13504688169172427]
```

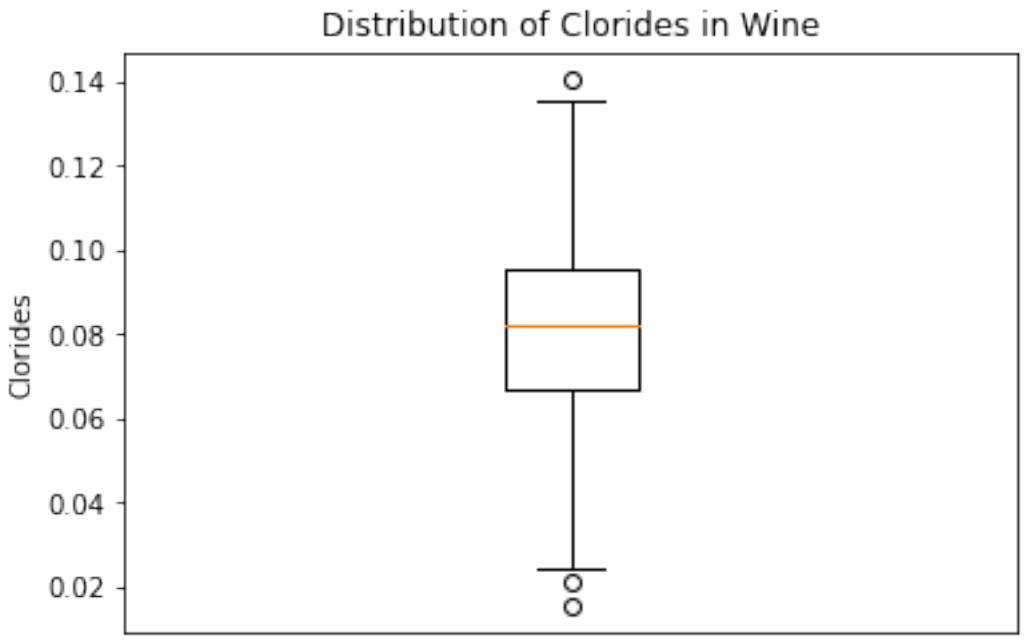


Histogram menunjukkan bahwa distribusi nilai *chlorides* dalam 1000 sampel anggur berbentuk *bell-shaped* atau memiliki distribusi normal. Walaupun begitu, jika dibandingkan dengan kolom lainnya, sekilas distribusi ini terlihat *negatively skewed*. Distribusi tersebut mencapai nilai puncak pada tingkat keasaman 0.089 - 0.095 (dengan frekuensi sebanyak 119). Nilai *chlorides* memiliki range sekitar 0.015 - 0.135. Distribusi ini memiliki beberapa nilai dengan tingkat sangat tinggi ataupun sangat rendah, namun tidak mempengaruhi bentuk distribusi.

[16]:
```
# ======= Boxplot ============
plt.boxplot(dataChlorides)

# Set attributes
plt.title('Distribution of Clorides in Wine')
plt.ylabel('Clorides')
plt.xticks([],[])

# Show graph
plt.show()
```



Berdasarkan visualisasi di atas, boxplot menunjukkan nilai minimum distribusi adalah sektar 0.02, sedangkan nilai maksimum terdapat pada 0.14. Nilai median *chlorides* berada pada 0.08, dengan *Interquartile Range* diantara 0.07 - 0.09. Terdapat tiga *outlier* pada distribusi, dua berada di bawah nilai minimum dan satu berada di atas nilai maksimum.

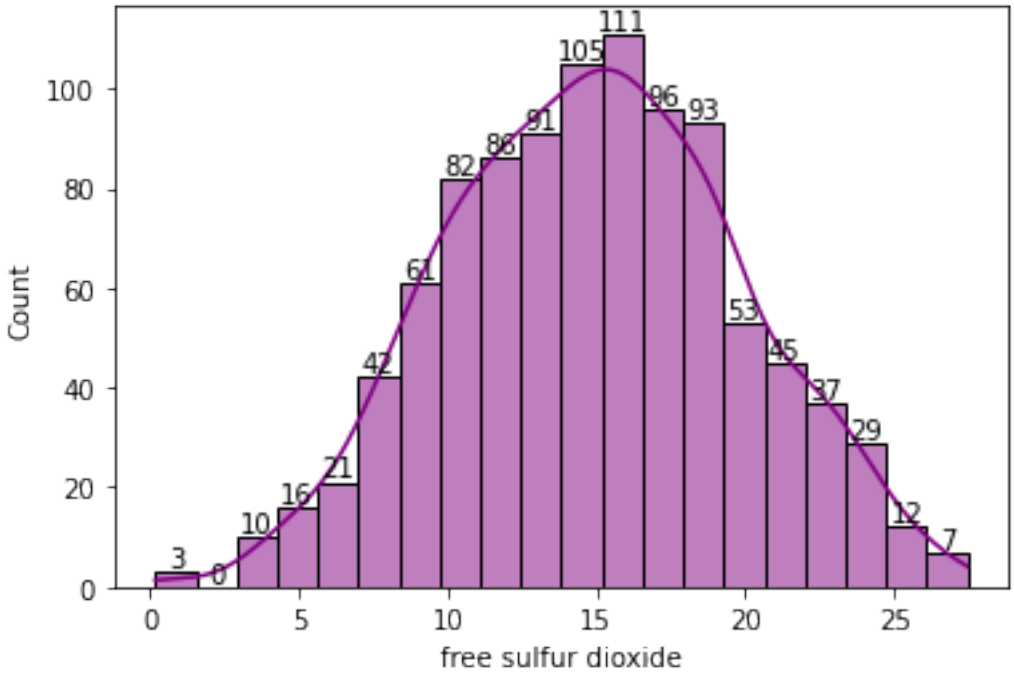
0.0.6 6. free sulfur dioxide

[17]:
```
# Data
dataFreeSulfurDioxide = dataAnggur['free sulfur dioxide']
```

[18]:
```
# ======= Histogram ============
ax = sns.histplot(dataFreeSulfurDioxide, color='purple', stat = 'count', kde =␣
 ↪True)
for i in ax.containers:
    ax.bar_label(i,)

# Print the bin edges
bin_edges = [patch.get_x() for patch in ax.patches]
print("Bin Edges (from leftmost): ", bin_edges)
```

```
Bin Edges (from leftmost):  [0.19467852332693703, 1.5580708683818822,
2.9214632134368275, 4.284855558491773, 5.648247903546719, 7.011640248601665,
8.37503259365661, 9.738424938711553, 11.1018172837665, 12.465209628821444,
13.828601973876392, 15.19199431893134, 16.55538666398628, 17.918779009041224,
19.28217135409617, 20.64556369915112, 22.00895604420606, 23.372348389261006,
24.735740734315954, 26.099133079370898]
```

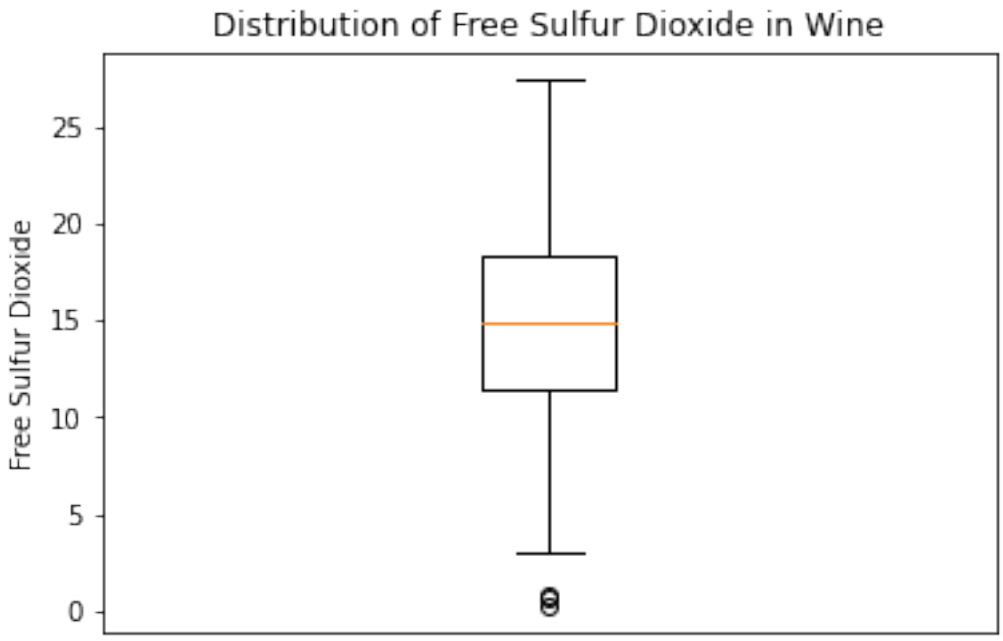


Histogram menunjukkan bahwa distribusi nilai *free sulfur dioxide* dalam 1000 sampel anggur berbentuk *bell-shaped* atau memiliki distribusi normal. Distribusi tersebut mencapai nilai puncak pada tingkat sulfur dioksida 15.19 - 16.55 (dengan frekuensi sebanyak 111). Nilai *free sulfur dioxide* memiliki range sekitar 0.194 - 26.099. Distribusi ini memiliki beberapa nilai dengan tingkat sangat tinggi ataupun sangat rendah, namun tidak mempengaruhi bentuk distribusi.

[19]:
```
# ======= Boxplot ============
plt.boxplot(dataFreeSulfurDioxide)

# Set attributes
plt.title('Distribution of Free Sulfur Dioxide in Wine')
plt.ylabel('Free Sulfur Dioxide')
plt.xticks([],[])

# Show graph
plt.show()
```



Berdasarkan visualisasi di atas, boxplot menunjukkan nilai minimum distribusi adalah sektar 2, sedangkan nilai maksimum terdapat pada 27. Nilai median *free sulfur dioxide* berada pada sekitar 15, dengan *Interquartile Range* diantara 10 - 17. Terdapat beberapa *outlier* pada distribusi, semua memiliki nilai di bawah minimum.

0.0.7 7. total sulfur dioxide

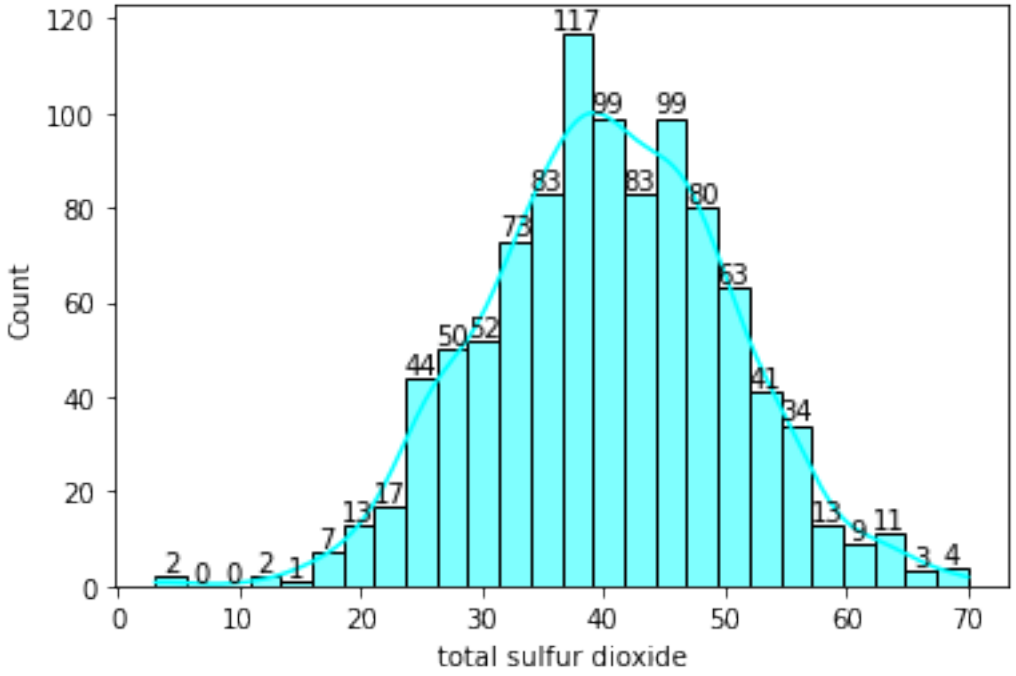
[20]:
```
# Data
dataTotalSulfurDioxide = dataAnggur['total sulfur dioxide']
```

[21]:
```
# ======= Histogram ============
ax = sns.histplot(dataTotalSulfurDioxide, color='cyan', stat = 'count', kde =␣
 ↪True)
for i in ax.containers:
    ax.bar_label(i,)

# Print the bin edges
bin_edges = [patch.get_x() for patch in ax.patches]
print("Bin Edges (from leftmost): ", bin_edges)
```

```
Bin Edges (from leftmost):  [3.1500000000000004, 5.719615384615384,
8.289230769230768, 10.858846153846155, 13.428461538461537, 15.998076923076923,
18.567692307692305, 21.137307692307687, 23.70692307692307, 26.276538461538458,
28.84615384615384, 31.41576923076922, 33.9853846153846, 36.55499999999999,
39.124615384615375, 41.69423076923076, 44.263846153846146, 46.83346153846153,
49.40307692307691, 51.9726923076923, 54.54230769230768, 57.11192307692306,
```

```
59.68153846153845, 62.25115384615384, 64.82076923076923, 67.3903846153846]
```

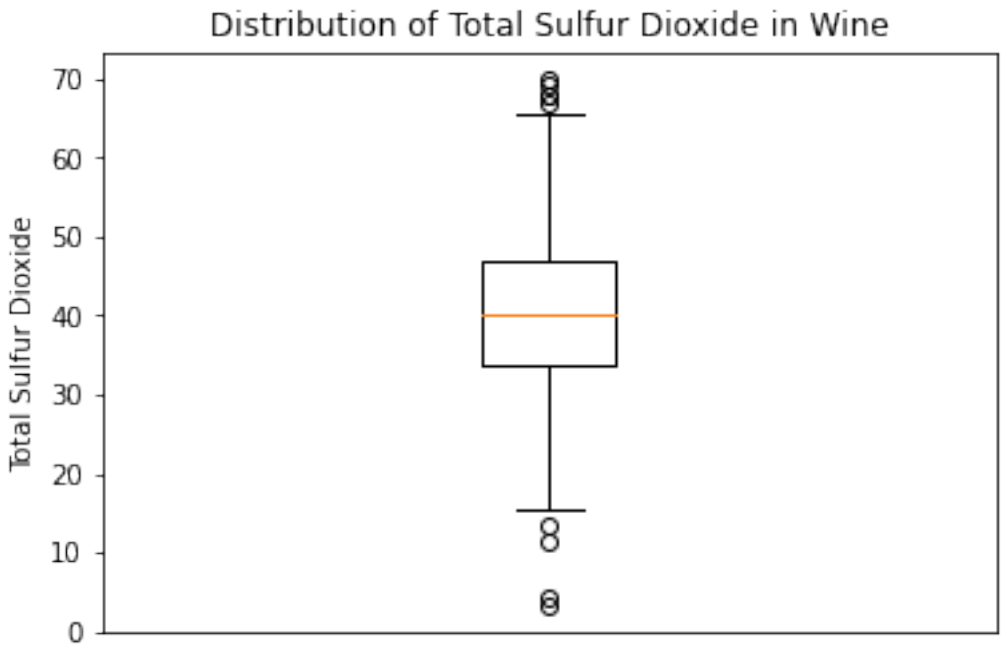


Histogram menunjukkan bahwa distribusi nilai *total sulfur dioxide* dalam 1000 sampel anggur berbentuk *bell-shaped* atau memiliki distribusi normal. Distribusi tersebut mencapai nilai puncak pada tingkat sulfur dioksida 36.55 - 39.12(dengan frekuensi sebanyak 117). Nilai *total sulfur dioxide* memiliki range sekitar 3.15 - 67.3. Distribusi ini memiliki beberapa nilai dengan tingkat sangat tinggi ataupun sangat rendah, namun tidak mempengaruhi bentuk distribusi.

[22]:
```
# ======= Boxplot ============
plt.boxplot(dataTotalSulfurDioxide)

# Set attributes
plt.title('Distribution of Total Sulfur Dioxide in Wine')
plt.ylabel('Total Sulfur Dioxide')
plt.xticks([],[])

# Show graph
plt.show()
```



Berdasarkan visualisasi di atas, boxplot menunjukkan nilai minimum distribusi *total sulfur dioxide* adalah sektar 15, sedangkan nilai maksimum terdapat pada 65. Nilai median *total sulfur dioxide* berada pada sekitar 40, dengan *Interquartile Range* diantara 35 - 45. Terdapat beberapa *outlier* pada distribusi, yang memiliki nilai di atas maksimum dan di bawah minimum.

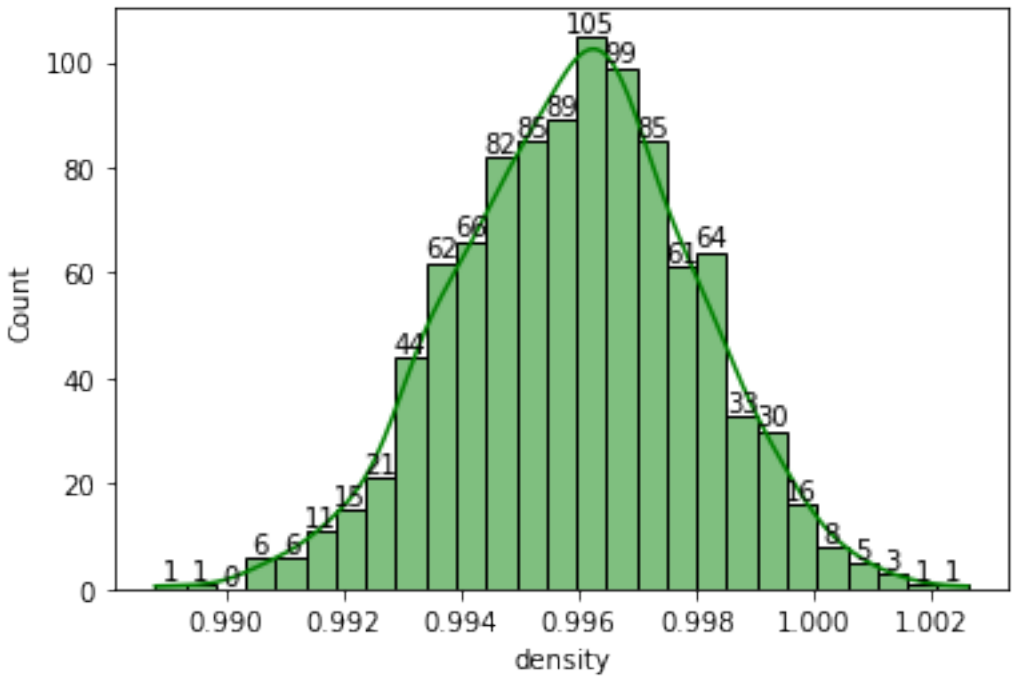
0.0.8 8. density

[23]:
```
# Data
dataDensity = dataAnggur['density']
```

[24]:
```
# ======= Histogram ============
ax = sns.histplot(dataDensity, color='green', stat = 'count', kde = True)
for i in ax.containers:
    ax.bar_label(i,)

# Print the bin edges
bin_edges = [patch.get_x() for patch in ax.patches]
print("Bin Edges (from leftmost): ", bin_edges)
```

```
Bin Edges (from leftmost):  [0.9888, 0.9893111111111111, 0.9898222222222224,
0.9903333333333333, 0.9908444444444444, 0.9913555555555555, 0.9918666666666667,
0.9923777777777778, 0.9928888888888889, 0.9934, 0.9939111111111111,
0.9944222222222222, 0.9949333333333333, 0.9954444444444446, 0.9959555555555555,
0.9964666666666666, 0.9969777777777777, 0.9974888888888889, 0.998,
0.998511111111111, 0.9990222222222221, 0.9995333333333334, 1.0000444444444443,
1.0005555555555556, 1.0010666666666665, 1.001577777777778, 1.0020888888888888]
```

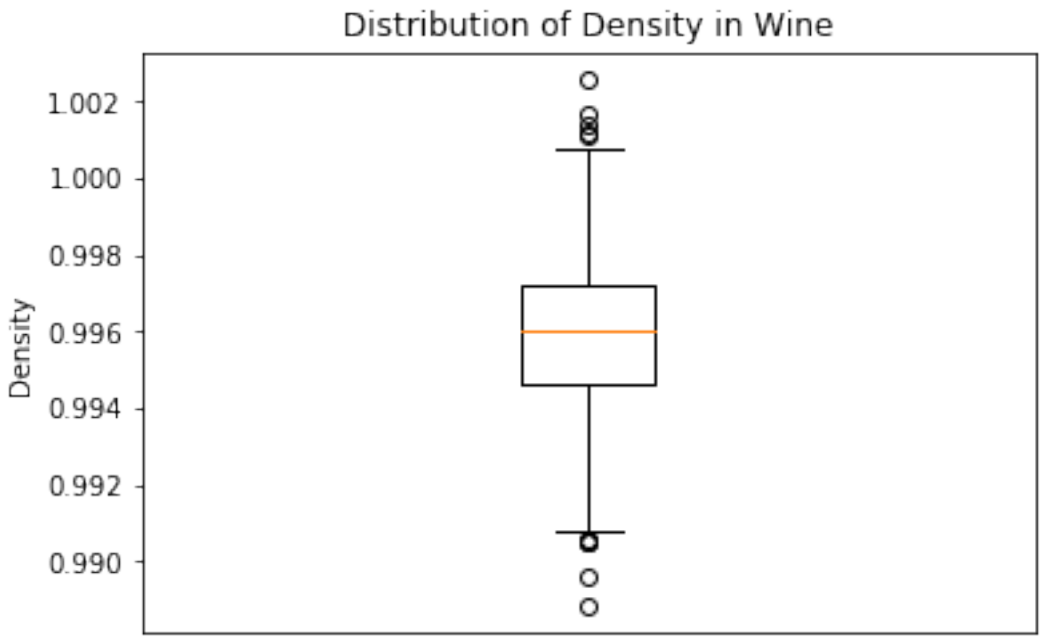


Histogram menunjukkan bahwa distribusi nilai *density* dalam 1000 sampel anggur berbentuk *bell-shaped* atau memiliki distribusi normal. Distribusi tersebut mencapai nilai puncak pada tingkat kepadatan 0.9959 - 0.9964 (dengan frekuensi sebanyak 105). Nilai *density* memiliki range sekitar 0.988 - 1.002. Distribusi ini memiliki beberapa nilai dengan tingkat sangat tinggi ataupun sangat rendah, namun tidak mempengaruhi bentuk distribusi.

[25]:
```
# ======= Boxplot ============
plt.boxplot(dataDensity)

# Set attributes
plt.title('Distribution of Density in Wine')
plt.ylabel('Density')
plt.xticks([],[])

# Show graph
plt.show()
```



Berdasarkan visualisasi di atas, boxplot menunjukkan nilai minimum distribusi *density* adalah sektar 0.991, sedangkan nilai maksimum terdapat pada 1.001. Nilai median *density* berada pada sekitar 0.996, dengan *Interquartile Range* diantara 0.995 - 0.997. Terdapat beberapa *outlier* pada distribusi, yang memiliki nilai di atas maksimum dan di bawah minimum.

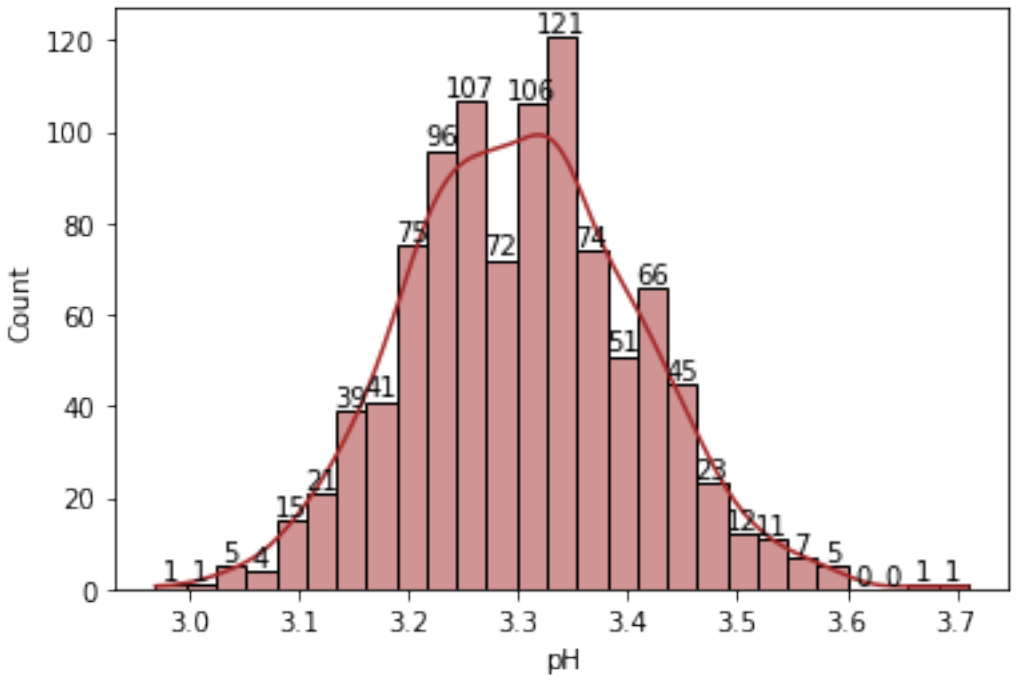
0.0.9 9. pH

[26]:
```
# Data
datapH = dataAnggur['pH']
```

[27]:
```
# ======= Histogram ============
ax = sns.histplot(datapH, color='brown', stat = 'count', kde = True)
for i in ax.containers:
    ax.bar_label(i,)

# Print the bin edges
bin_edges = [patch.get_x() for patch in ax.patches]
print("Bin Edges (from leftmost): ", bin_edges)
```

```
Bin Edges (from leftmost):  [2.97, 2.9974074074074073, 3.024814814814815,
3.0522222222222224, 3.07962962962963, 3.107037037037037, 3.1344444444444446,
3.1618518518518517, 3.1892592592592592, 3.216666666666667, 3.2440740740740743,
3.2714814814814814, 3.298888888888889, 3.326296296296296, 3.3537037037037036,
3.381111111111111, 3.4085185185185187, 3.435925925925926, 3.463333333333333,
3.4907407407407405, 3.518148148148148, 3.5455555555555556, 3.572962962962963,
3.60037037037037, 3.6277777777777773, 3.655185185185185, 3.6825925925925924]
```

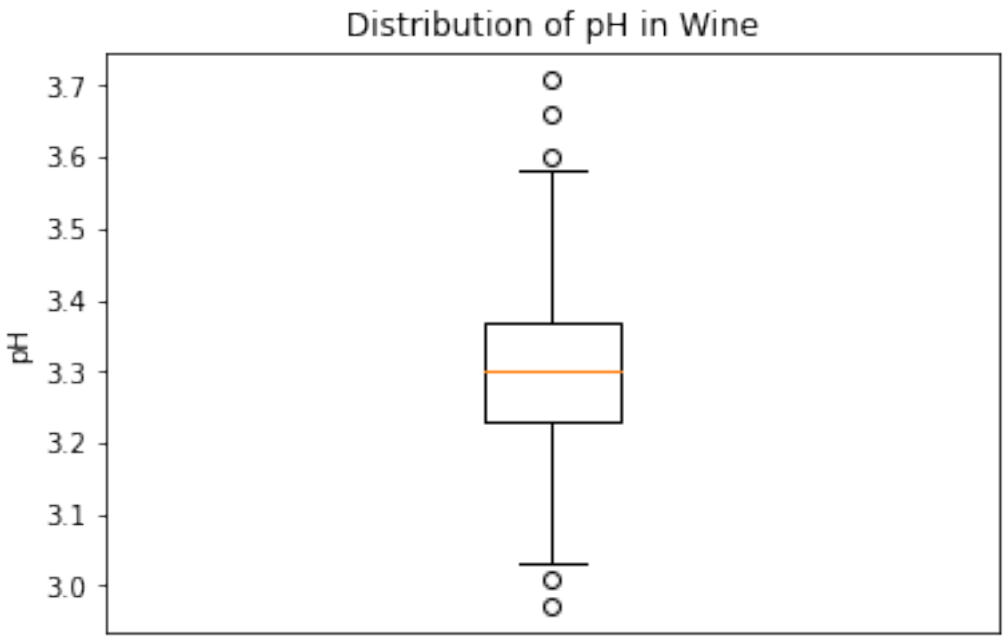


Histogram menunjukkan distribusi nilai *pH* dalam sampel 1000 anggur. Distribusi tersebut terlihat berbentuk *bell-shaped*, dengan distribusi normal. Walaupun begitu, jika dibandingkan dengan histogram kolom lainnya, bentuk ini sekilas terlihat lebih *positively skewed*. Distribusi ini memiliki nilai puncak pada range pH 3.32 - 3.35, dengan frekuensi 121. Nilai *pH* berkisar antara 2.97 - 3.68.

[28]:
```
# ======= Boxplot ============
plt.boxplot(datapH)

# Set attributes
plt.title('Distribution of pH in Wine')
plt.ylabel('pH')
plt.xticks([],[])

# Show graph
plt.show()
```



Berdasarkan visualisasi di atas, boxplot menunjukkan nilai minimum distribusi *pH* adalah sektar 3.05, sedangkan nilai maksimum terdapat pada 3.58. Nilai median *pH* berada pada sekitar 3.3, dengan *Interquartile Range* diantara 3.25 - 3.35 Terdapat beberapa *outlier* pada distribusi, yang memiliki nilai di atas maksimum dan di bawah minimum.

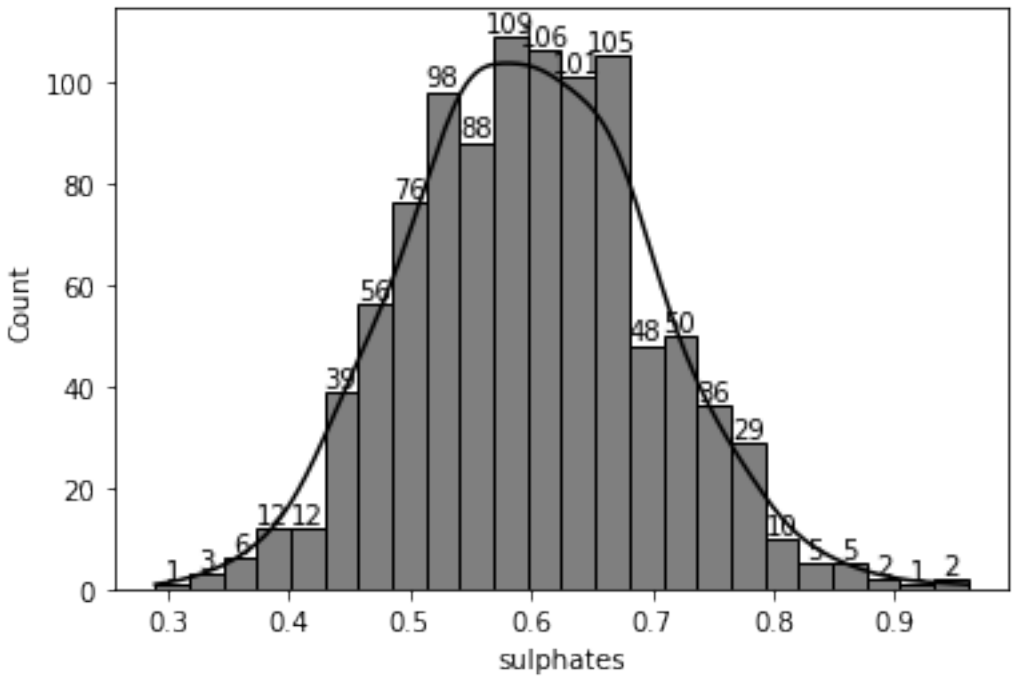
0.0.10 10. sulphates

[29]:
```
# Data
dataSulphates = dataAnggur['sulphates']
```

[30]:
```
# ======= Histogram ============
ax = sns.histplot(dataSulphates, color='black', stat = 'count', kde = True)
for i in ax.containers:
    ax.bar_label(i,)

# Print the bin edges
bin_edges = [patch.get_x() for patch in ax.patches]
print("Bin Edges (from leftmost): ", bin_edges)
```

```
Bin Edges (from leftmost):  [0.29000000000000004, 0.3179166666666666,
0.3458333333333333, 0.37375, 0.4016666666666666, 0.4295833333333333, 0.4575,
0.4854166666666666, 0.5133333333333334, 0.54125, 0.5691666666666666,
0.5970833333333332, 0.625, 0.6529166666666666, 0.6808333333333332, 0.70875,
0.7366666666666666, 0.7645833333333332, 0.7925, 0.8204166666666666,
0.8483333333333332, 0.87625, 0.9041666666666666, 0.9320833333333332]
```

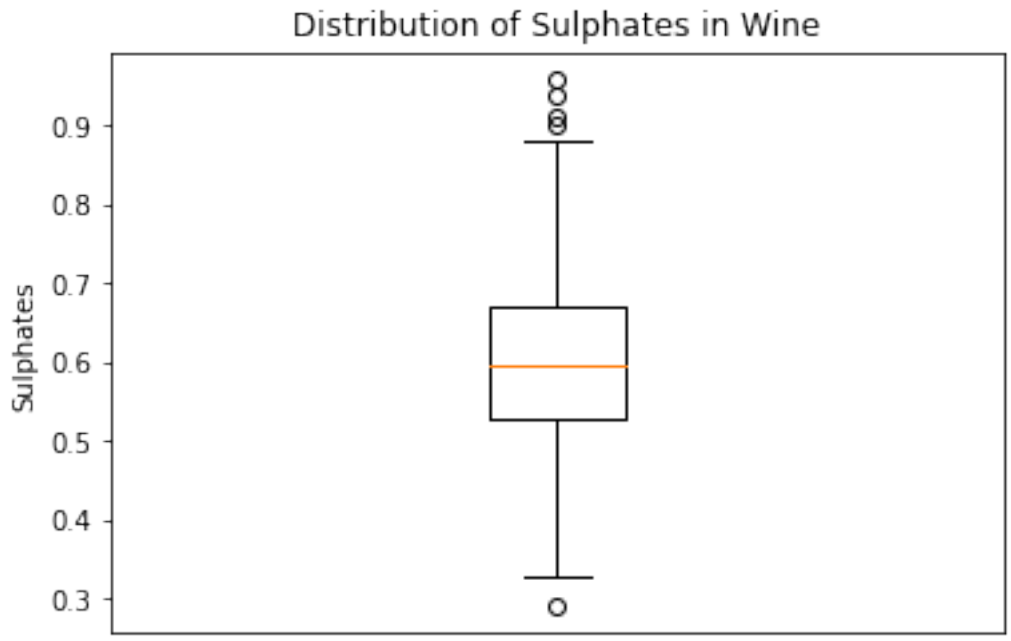


Histogram menunjukkan bahwa distribusi nilai *sulphates* dalam 1000 sampel anggur berbentuk *bell-shaped* atau memiliki distribusi normal. Distribusi tersebut mencapai nilai puncak pada tingkat kepadatan 0.569 - 0.597 (dengan frekuensi sebanyak 109). Nilai *sulphates* memiliki range sekitar 0.29 - 0.96. Distribusi ini memiliki beberapa nilai dengan tingkat sangat tinggi ataupun sangat rendah, namun tidak mempengaruhi bentuk distribusi.

[31]:
```
# ======= Boxplot ============
plt.boxplot(dataSulphates)

# Set attributes
plt.title('Distribution of Sulphates in Wine')
plt.ylabel('Sulphates')
plt.xticks([],[])

# Show graph
plt.show()
```



Berdasarkan visualisasi di atas, boxplot menunjukkan nilai minimum distribusi *sulphates* adalah sektar 0.33, sedangkan nilai maksimum terdapat pada 0.87. Nilai median *sulphates* berada pada sekitar 0.6, dengan *Interquartile Range* diantara 0.55 - 0.65. Terdapat beberapa *outlier* pada distribusi, sebagian besar memiliki nilai di atas nilai maksimum.

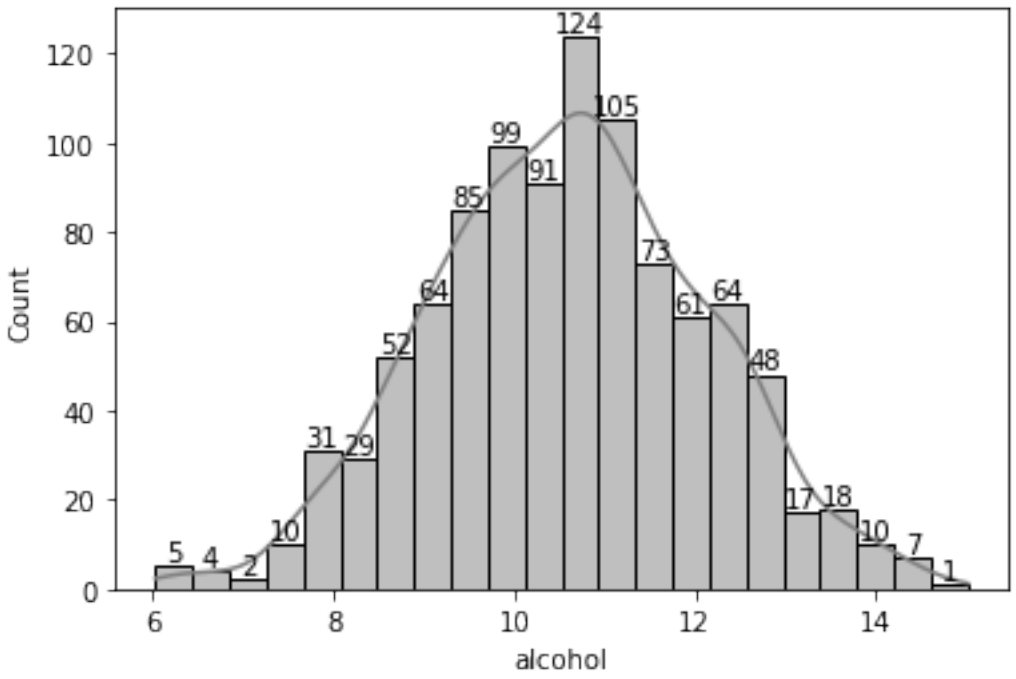
0.0.11 11. alcohol

[32]:
```
# Data
dataAlcohol = dataAnggur['alcohol']
```

[33]:
```
# ======= Histogram ============
ax = sns.histplot(dataAlcohol, color='grey', stat = 'count', kde = True)
for i in ax.containers:
    ax.bar_label(i,)

# Print the bin edges
bin_edges = [patch.get_x() for patch in ax.patches]
print("Bin Edges (from leftmost): ", bin_edges)
```

```
Bin Edges (from leftmost):  [6.03, 6.438636363636363, 6.847272727272728,
7.255909090909091, 7.664545454545454, 8.07318181818182, 8.48181818181818,
8.890454545454546, 9.29909090909091, 9.707727272727272, 10.116363636363637,
10.524999999999999, 10.933636363636364, 11.342272727272729, 11.75090909090909,
12.159545454545455, 12.568181818181817, 12.976818181818182, 13.385454545454547,
13.794090909090908, 14.202727272727273, 14.611363636363635]
```

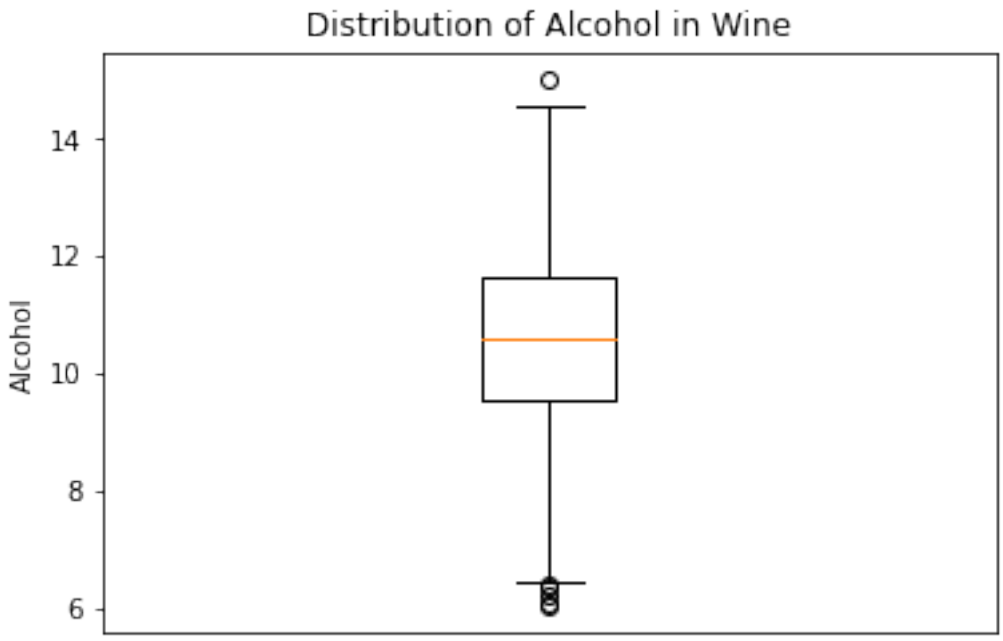


Histogram menunjukkan bahwa distribusi nilai *alcohol* dalam 1000 sampel anggur berbentuk *bell-shaped* atau memiliki distribusi normal. Distribusi tersebut mencapai nilai puncak pada tingkat kepadatan 10.52 - 10.93 (dengan frekuensi sebanyak 124). Nilai *alcohol* memiliki range sekitar 6.03 - 14.61. Distribusi ini memiliki beberapa nilai dengan tingkat sangat tinggi ataupun sangat rendah, namun tidak mempengaruhi bentuk distribusi.

[34]:
```
# ======= Boxplot ============
plt.boxplot(dataAlcohol)

# Set attributes
plt.title('Distribution of Alcohol in Wine')
plt.ylabel('Alcohol')
plt.xticks([],[])

# Show graph
plt.show()
```



Berdasarkan visualisasi di atas, boxplot menunjukkan nilai minimum distribusi *alcohol* adalah sektar 6.5, sedangkan nilai maksimum terdapat pada 14.5. Nilai median *alcohol* berada pada sekitar 10.5, dengan *Interquartile Range* diantara 9.55 - 11.5. Terdapat beberapa *outlier* pada distribusi, sebagian besar memiliki nilai di bawah nilai minimum.

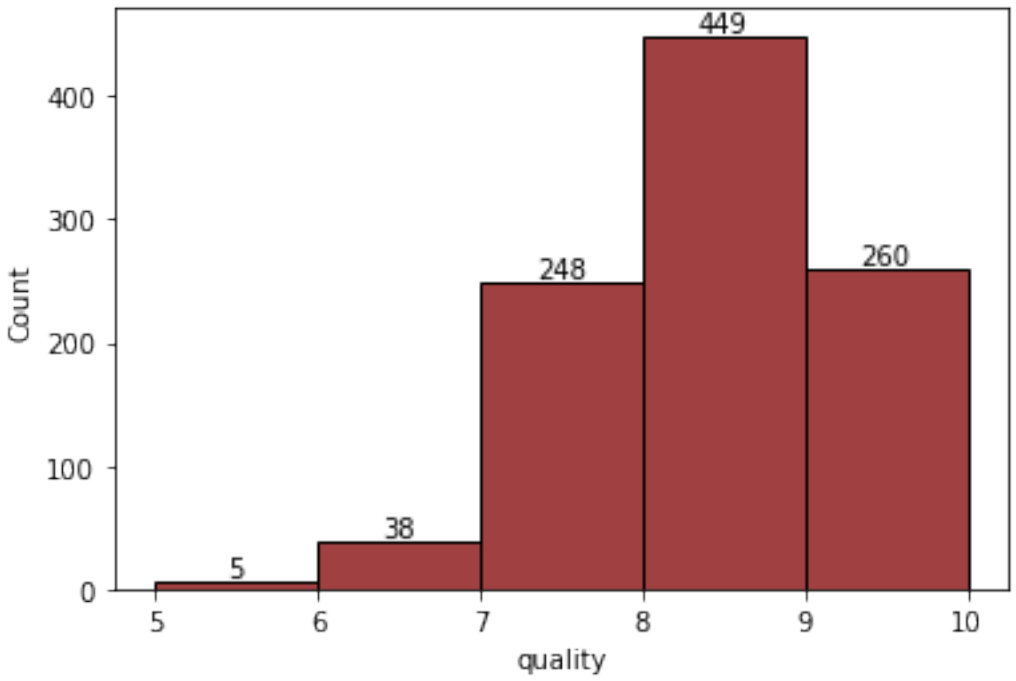
0.0.12 12. quality

[35]:
```
# Data
dataQuality = dataAnggur['quality']
```

[36]:
```
# ======= Histogram ============
ax = sns.histplot(dataQuality, color='maroon', stat = 'count', bins=5)
for i in ax.containers:
    ax.bar_label(i,)

# Print the bin edges
bin_edges = [patch.get_x() for patch in ax.patches]
print("Bin Edges (from leftmost): ", bin_edges)
```

```
Bin Edges (from leftmost):  [5.0, 6.0, 7.0, 8.0, 9.0]
```



Histogram menunjukkan bahwa distribusi nilai *quality* dalam 1000 sampel anggur. Data pada kolom ini bertipe diskrit. Distribusi tersebut mencapai nilai puncak pada tingkat kualitas 8 - 9 (dengan frekuensi sebanyak 449). Nilai *quality* memiliki range sekitar 5 - 10. Distribusi ini memiliki beberapa nilai dengan tingkat sangat tinggi ataupun sangat rendah, namun tidak mempengaruhi bentuk distribusi.

[37]:
```
# ======= Boxplot ============
plt.boxplot(dataQuality)

# Set attributes
plt.title('Distribution of Quality in Wine')
plt.ylabel('Quality')
plt.xticks([],[])

# Show graph
plt.show()
```

Berdasarkan visualisasi di atas, boxplot menunjukkan nilai minimum distribusi *quality* adalah 5, sedangkan nilai maksimum terdapat pada 10. Nilai median *quality* berada pada nilai 8, dengan *Interquartile Range* diantara 7 - 9. Tidak terdapat *outlier* pada distribusi.

3

April 18, 2023

[1]:
```
import pandas as pd
import scipy.stats as st
import seaborn as sns
from scipy import stats
import random

dataAnggur = pd.read_csv('../data/anggur.csv')
```

Menentukan setiap kolom numerik berdistribusi normal atau tidak. Gunakan normality test yang dikaitkan dengan histogram plot

0.1 Asumsi

- Suatu distibusi dapat dikatakan *positively skewed* apabila memiliki nilai *skewness* lebih besar dari 0.5
- Suatu distibusi dapat dikatakan *negatively skewed* apabila memiliki nilai *skewness* lebih besar dari -0.5
- Suatu distibusi dapat dikatakan simetris apabila memiliki nilai *skewness* diantara -0.5 sampai 0.5
- Jika nilai modus = median = mean maka nilai *skewness* adalah nol
- Jika nilai modus < median < mean maka distribusi dapat dikatakan *positively skewed*
- Jika nilai modus > median > mean maka distribusi dapat dikatakan *negatively skewed*

0.2 Metode Pengetesan Normalitas

- Pengetesan melalui grafik. Pengetesan dilakukan dengan membandingkan grafik (histogram, QQ plot, dll.) dengan grafik yang bersesuaian dari sampel data yang berdistribusi normal. Pendekatan ini bersifat informal.
- Pengetesan statistik. Pengetesan ini dilakukan dengan uji hipotesis. Contoh pengetesan normalitas secara statistik adalah Shapiro-Wilk test, Kolmogorov-Smirnov test, dan Jarque-Bera test. Pengetesan ini bersifat formal.

Pada bagian ini, pengetesan normalitas yang digunakan yaitu pengetesan melalui histogram, yang akan dikaitkan dengan statistik-statistik tertentu seperti *skewness*, *excess kurtosis*, mean, median, dan modus. Lalu, hasil pengetesan tersebut dibandingkan dengan hasil pengetesan normalitas statistik, yaitu D'Agostino-Pearson Test dan Shapiro-Wilk Test. Berikut adalah penjelasan mengenai pengetesan yang digunakan.

0.2.1 1. Pengetesan melalui Histogram

- Histogram distribusi normal berbentuk bell-shaped yang simetris.
- Histogram distribusi normal berbentuk simetris terhadap sumbu tegak x = mean.
- Kurva distribusi normal mendekati sumbu datar secara asimtotik ke kiri dan kanan.
- Skewness distribusi normal bernilai 0, yang menandakan distribusi simetris.
- Kurtosis distribusi normal bernilai 3 (excess kurtosis bernilai 0).

Pada pengetesan yang dilakukan, histogram setiap kolom divisualisasikan dan dibandingkan dengan ciri histogram di atas. Namun, untuk perhitungan skewness dan kurtosis, diberikan toleransi sebesar 0.5, seperti asumsi yang dituliskan di atas.

0.2.2 2. D'Agostino-Pearson Test

D'Agostino-Pearson Test adalah tes normalitas yang perhitungannya menggabungkan hasil tes skewness dan kurtosis D'Agostino. Tes ini kurang sensitif terhadap penyimpangan (deviasi) dari distribusi normal di ekor distribusi.

$$K^2 = Z_s^2 + Z_k^2$$

Z_s^2 adalah z-score dari tes skewness D'Agostino dan Z_k^2 adalah z-score dari tes kurtosis D'Agostino. Jika hipotesis null terbukti, K^2 diaproksimasi terdistribusi chi-squared dengan derajat kebebasan 2.

Mekanisme Pengujian

- H_0
  Data berdistribusi normal.
- H_1
  Data tidak berdistribusi normal.
- Tingkat Signifikansi
  $\alpha = 0.05$
- Penarikan Kesimpulan dengan Tes Signifikansi
  Jika $p > \alpha$, maka H_0 *fail to reject*, artinya data berdistribusi normal. Sebaliknya, jika $p , maka H_0 *rejected*, artinya data tidak berdistribusi normal.

0.2.3 3. Shapiro-Wilk Test

Shapiro-Wilk Test adalah tes normalitas yang perhitungannya didasari oleh perbandingan antara data yang diobservasi dan *expected normal distribution* dari data tersebut. Tes ini tidak terlalu *reliable* untuk jumlah sampel yang kecil. Namun, sampel yang ada pada tiap kolom sudah cukup besar, yaitu sebanyak 1000 sampel.

Mekanisme Pengujian

- H_0
  Data berdistribusi normal.
- H_1
  Data tidak berdistribusi normal.
- Tingkat Signifikansi
  $\alpha = 0.05$

- Penarikan Kesimpulan dengan Tes Signifikansi
  Jika $p > \alpha$, maka H_0 *fail to reject*, artinya data berdistribusi normal. Sebaliknya, jika $p , maka H_0 *rejected*, artinya data tidak berdistribusi normal.

0.2.4 Fungsi Wrapper Normality Test

[2]:
```
def normalityTests(colName):
    colors = ['red', 'blue', 'pink', 'purple', 'black', 'green', 'orange',␣
 ↪'magenta']
    dataCol = dataAnggur[colName]

    # Create histogram
    ax = sns.histplot(dataCol, color=colors[random.randint(0, len(colors) -␣
 ↪1)], stat = 'count', kde = True)
    for i in ax.containers:
        ax.bar_label(i,)

    # Check skewness
    print("Skewness        :" , dataCol.skew())

    # Check kurtosis
    print("Excess Kurtosis :", dataCol.kurtosis())

    # Jarque bera Test
    print("\nD'Agustino-Pearson Test")
    stat, p = st.normaltest(dataCol)
    print("Test            :", stat)
    print("Nilai p         :", p)

    # Shapiro-Wilk Test
    print("\nShapiro-Wilk Test")
    stat, p = stats.shapiro(dataCol)
    print("Test            :", stat)
    print("Nilai p         :", p)
```

0.3 Hasil Tes Normalitas

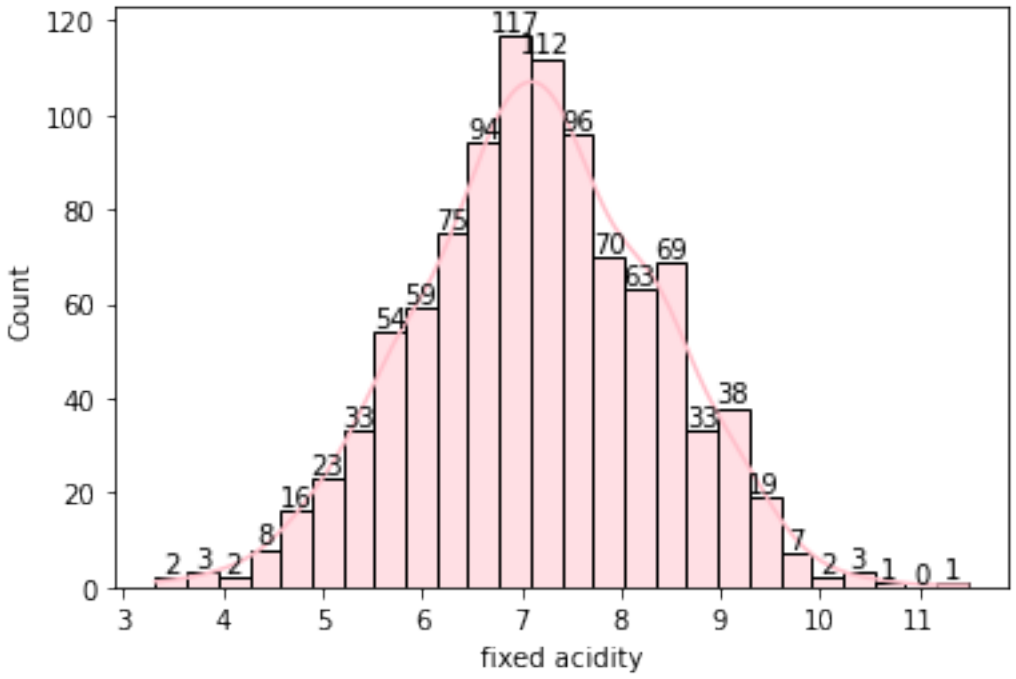
0.3.1 1. fixed acidity

[3]:
```
# FIXED ACIDITY
normalityTests("fixed acidity")
```

```
Skewness        : -0.028878575532660055
Excess Kurtosis : -0.019292120932933532

D'Agustino-Pearson Test
Test            : 0.14329615661430725
Nilai p         : 0.9308584274486692
```

```
Shapiro-Wilk Test
Test             : 0.9990411400794983
Nilai p          : 0.8935267925262451
```



Hasil Tes

- Berdasarkan histogram di atas, kolom "fixed acidity" dapat dianggap berdistribusi normal karena histogramnya berbentuk *bell-shaped* yang simetris. *Skewness*-nya juga berada di antara -0.5 dan 0.5 (*skewness* = -0.0288, -0.5 < -0.0288 < 0.5), yang menandakan bahwa histogram di atas simetris. *Excess kurtosis*-nya juga berada di antara -0.5 dan 0.5 (*excess kurtosis* = -0.0192, -0.5 < -0.0192 < 0.5), yang menandakan bahwa histogram di atas memiliki keruncingan distribusi normal.
- Berdasarkan D'Agustino-Pearson Test, kolom "fixed acidity" dapat dianggap berdistribusi normal karena Nilai P-nya lebih dari 0.05 (Nilai P = 0.9308 > 0.05).
- Berdasarkan Shapiro-Wilk Test, kolom "fixed acidity" dapat dianggap berdistribusi normal karena Nilai P-nya lebih dari 0.05 (Nilai P = 0.8935 > 0.05).

Kesimpulan

Berdasarkan tes normalitas yang dilakukan di atas, kolom "fixed acidity" berdistribusi normal.

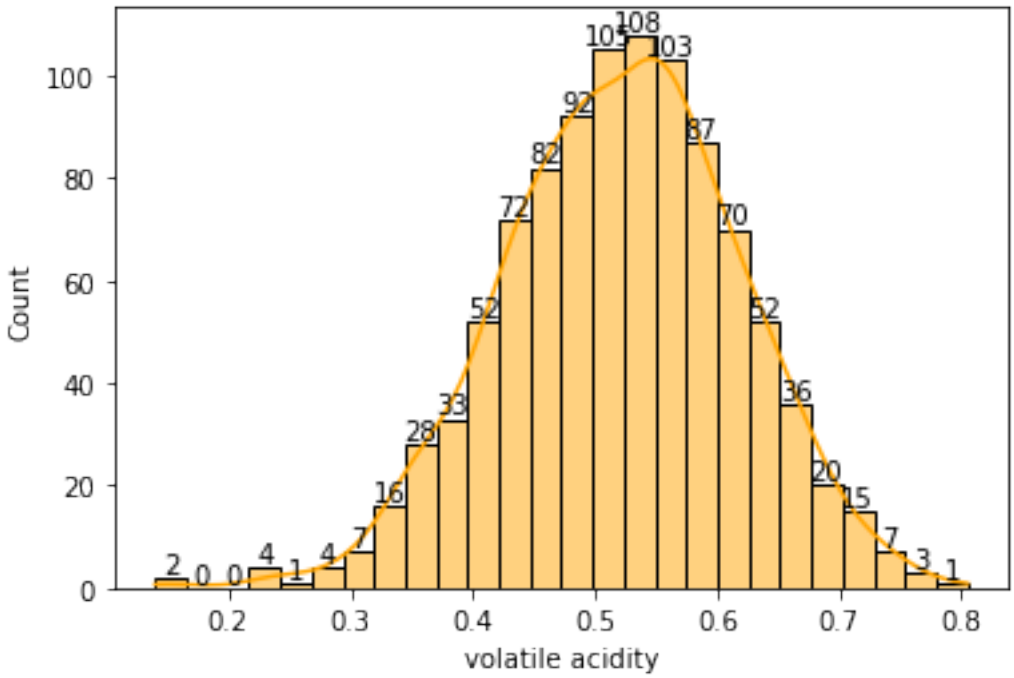
0.3.2 2. volatile acidity

[4]:
```
normalityTests("volatile acidity")
```

```
Skewness        : -0.1976986986092083
Excess Kurtosis : 0.16185290336961788

D'Agustino-Pearson Test
Test            : 7.581251985533493
Nilai p         : 0.022581461594113835

Shapiro-Wilk Test
Test            : 0.997028648853302
Nilai p         : 0.05993043631315231
```



Hasil Tes

- Berdasarkan histogram di atas, kolom "volatile acidity" dapat dianggap berdistribusi normal karena histogramnya berbentuk *bell-shaped* yang simetris. Walaupun begitu, sekilas histogramnya terlihat *negatively skewed*. *Skewness*-nya juga berada di antara -0.5 dan 0.5 (*skewness* = -0.1976, -0.5 < -0.1976 < 0.5), yang menandakan bahwa histogram di atas simetris. *Excess kurtosis*-nya juga berada di antara -0.5 dan 0.5 (*excess kurtosis* = 0.1618, -0.5 < 0.1618 < 0.5), yang menandakan bahwa histogram di atas memiliki keruncingan distribusi normal.
- Berdasarkan D'Agustino-Pearson Test, kolom "volatile acidity" tidak dapat dianggap berdistribusi normal karena Nilai P-nya tidak lebih dari 0.05 (Nilai P = 0.0225 <= 0.05).
- Berdasarkan Shapiro-Wilk Test, kolom "volatile acidity" dapat dianggap berdistribusi normal karena Nilai P-nya lebih dari 0.05 (Nilai P = 0.0599 > 0.05).

Terdapat perbedaan hasil tes diantara D'Agustino-Pearson Test dan Shapiro-Wilk Test karena nilai

P dianggap berada dalam perbatasan normalitas.

Kesimpulan
Berdasarkan tes normalitas yang dilakukan di atas, kolom "fixed acidity" tidak berdistribusi normal.

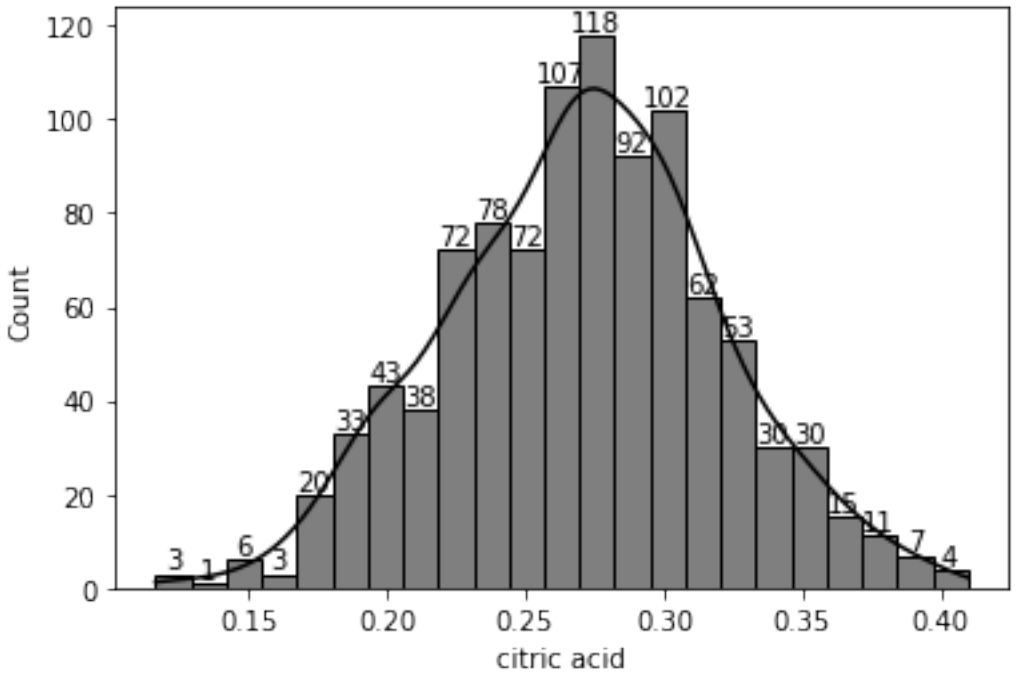
0.3.3 3. citric acid

[5]:
```
normalityTests("citric acid")
```

```
Skewness        : -0.045576058685017296
Excess Kurtosis : -0.1046792495951605

D'Agustino-Pearson Test
Test            : 0.7663607229418252
Nilai p         : 0.6816899375976969

Shapiro-Wilk Test
Test            : 0.9979573488235474
Nilai p         : 0.26522907614707947
```



Hasil Tes

- Berdasarkan histogram di atas, kolom "citric acid" dapat dianggap berdistribusi normal karena histogramnya berbentuk *bell-shaped* yang simetris. *Skewness*-nya juga berada di antara -0.5 dan 0.5 (*skewness* = -0.0455, -0.5 < -0.0455 < 0.5), yang menandakan bahwa histogram di atas simetris. *Excess kurtosis*-nya juga berada di antara -0.5 dan 0.5 (*excess*

kurtosis = -0.1046, -0.5 < -0.1046 < 0.5), yang menandakan bahwa histogram di atas memiliki keruncingan distribusi normal.

- Berdasarkan D'Agustino-Pearson Test, kolom "citric acid" dapat dianggap berdistribusi normal karena Nilai P-nya lebih dari 0.05 (Nilai P = 0.6816 > 0.05).
- Berdasarkan Shapiro-Wilk Test, kolom "citric acid" dapat dianggap berdistribusi normal karena Nilai P-nya lebih dari 0.05 (Nilai P = 0.2652 > 0.05).

Kesimpulan
Berdasarkan tes normalitas yang dilakukan di atas, kolom "citric acid" berdistribusi normal.

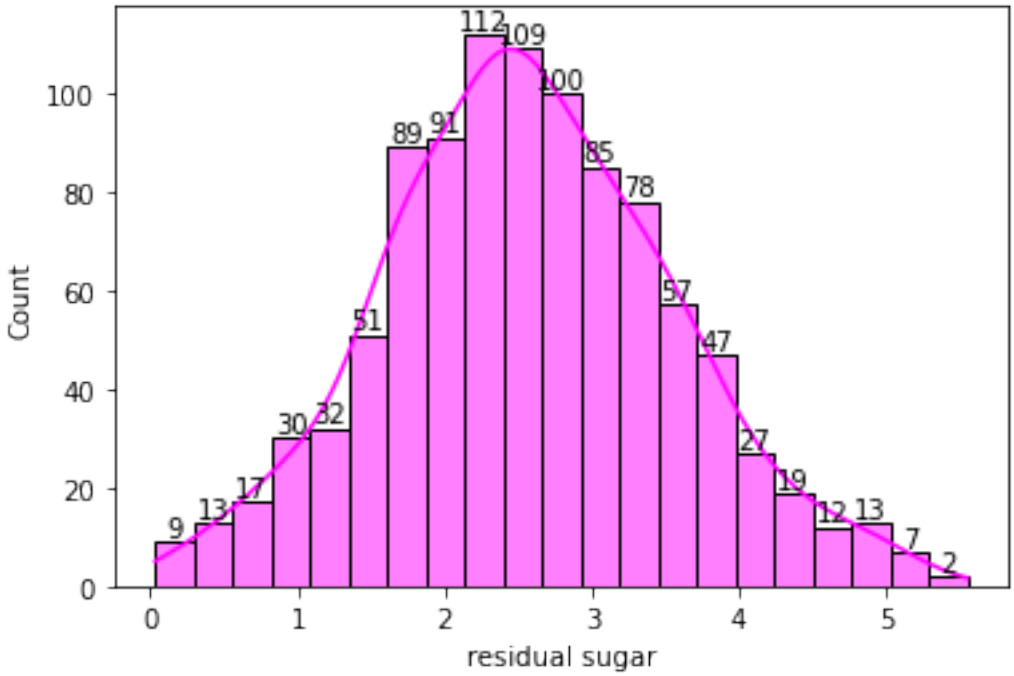
0.3.4 4. residual sugar

[6]:
```
normalityTests("residual sugar")
```

```
Skewness        : 0.13263808618992312
Excess Kurtosis : -0.04298003436476261

D'Agustino-Pearson Test
Test            : 2.9862716504538622
Nilai p         : 0.22466703321310558

Shapiro-Wilk Test
Test            : 0.9968547224998474
Nilai p         : 0.044918645173311234
```



Hasil Tes

- Berdasarkan histogram di atas, kolom "residual sugar" dapat dianggap berdistribusi normal karena histogramnya berbentuk *bell-shaped* yang simetris. *Skewness*-nya juga berada di antara -0.5 dan 0.5 (*skewness* = 0.1326, -0.5 < 0.1326 < 0.5), yang menandakan bahwa histogram di atas simetris. *Excess kurtosis*-nya juga berada di antara -0.5 dan 0.5 (*excess kurtosis* = -0.0429, -0.5 < -0.0429 < 0.5), yang menandakan bahwa histogram di atas memiliki keruncingan distribusi normal.
- Berdasarkan D'Agustino-Pearson Test, kolom "residual sugar" dapat dianggap berdistribusi normal karena Nilai P-nya lebih dari 0.05 (Nilai P = 0.2246 > 0.05).
- Berdasarkan Shapiro-Wilk Test, kolom "residual sugar" tidak dapat dianggap berdistribusi normal karena Nilai P-nya tidak lebih dari 0.05 (Nilai P = 0.0449 <= 0.05).

Terdapat perbedaan hasil tes diantara D'Agustino-Pearson Test dan Shapiro-Wilk Test karena nilai P dianggap berada dalam perbatasan normalitas.

Kesimpulan
Berdasarkan tes normalitas yang dilakukan di atas, kolom "residual sugar" berdistribusi normal.

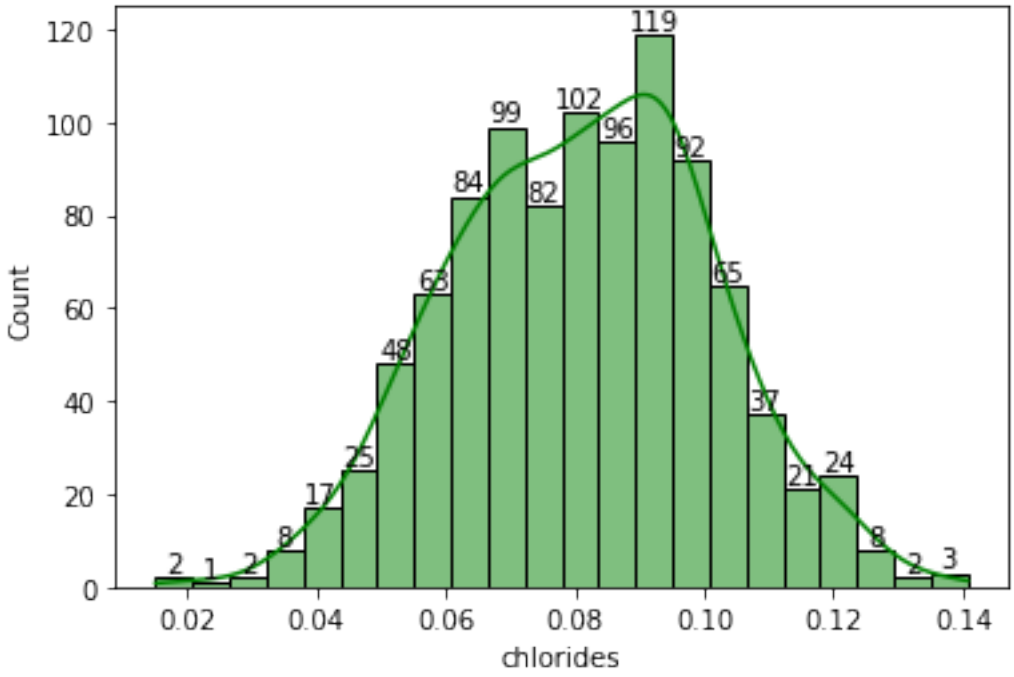
0.3.5 5. chlorides

[7]:
```
normalityTests("chlorides")
```

```
Skewness        : -0.05131929742072573
Excess Kurtosis : -0.2465081359240382

D'Agustino-Pearson Test
Test            : 3.538242355484952
Nilai p         : 0.17048274704296862

Shapiro-Wilk Test
Test            : 0.9976862072944641
Nilai p         : 0.17465530335903168
```



Hasil Tes

- Berdasarkan histogram di atas, kolom "chlorides" dapat dianggap berdistribusi normal karena histogramnya berbentuk *bell-shaped* yang simetris. *Skewness*-nya juga berada di antara -0.5 dan 0.5 (*skewness* = -0.0513, -0.5 < -0.0513 < 0.5), yang menandakan bahwa histogram di atas simetris. *Excess kurtosis*-nya juga berada di antara -0.5 dan 0.5 (*excess kurtosis* = -0.2465, -0.5 < -0.2465 < 0.5), yang menandakan bahwa histogram di atas memiliki keruncingan distribusi normal.
- Berdasarkan D'Agustino-Pearson Test, kolom "chlorides" dapat dianggap berdistribusi normal karena Nilai P-nya lebih dari 0.05 (Nilai P = 0.1704 > 0.05).
- Berdasarkan Shapiro-Wilk Test, kolom "chlorides" dapat dianggap berdistribusi normal karena Nilai P-nya lebih dari 0.05 (Nilai P = 0.1746 > 0.05).

Kesimpulan
Berdasarkan tes normalitas yang dilakukan di atas, kolom "chlorides" berdistribusi normal.

0.3.6 6. free sulfur dioxide

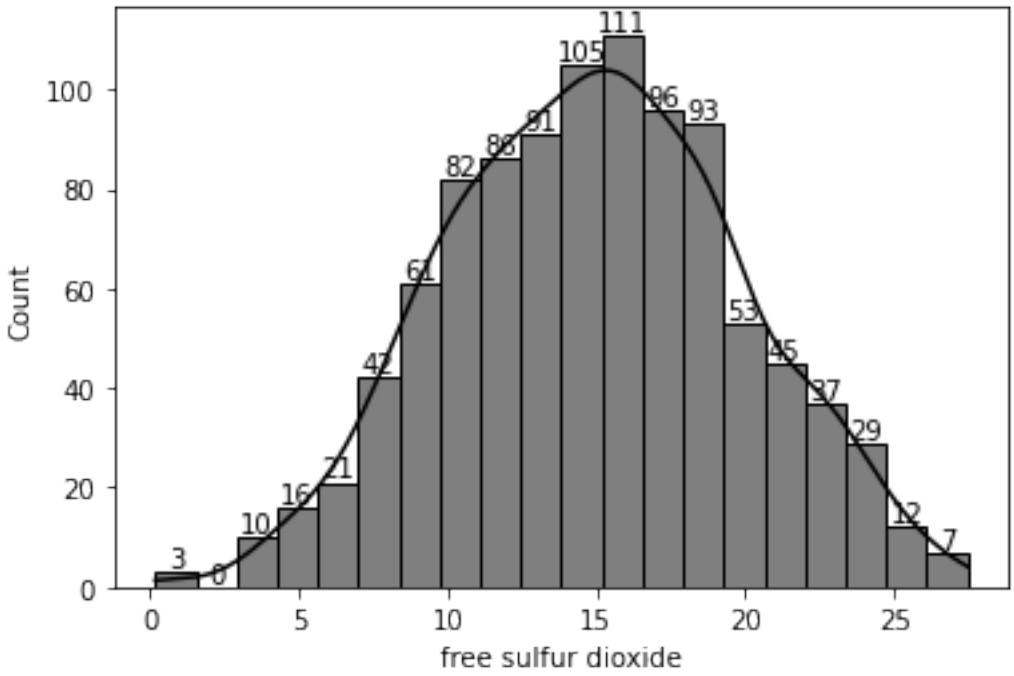
[8]:
```
normalityTests("free sulfur dioxide")
```

```
Skewness        : 0.007130415991143398
Excess Kurtosis : -0.36496364342685306

D'Agustino-Pearson Test
Test            : 8.099074980855514
```

```
Nilai p         : 0.01743043451827735

Shapiro-Wilk Test
Test            : 0.9968221783638
Nilai p         : 0.04255827143788338
```



Hasil Tes

- Berdasarkan histogram di atas, kolom "free sulfur dioxide" dapat dianggap berdistribusi normal karena histogramnya berbentuk *bell-shaped* yang simetris. *Skewness*-nya juga berada di antara -0.5 dan 0.5 (*skewness* = 0.0071, -0.5 < 0.0071 < 0.5), yang menandakan bahwa histogram di atas simetris. *Excess kurtosis*-nya juga berada di antara -0.5 dan 0.5 (*excess kurtosis* = -0.3649, -0.5 < -0.3649 < 0.5), yang menandakan bahwa histogram di atas memiliki keruncingan distribusi normal.
- Berdasarkan D'Agustino-Pearson Test, kolom "free sulfur dioxide" tidak dapat dianggap berdistribusi normal karena Nilai P-nya tidak lebih dari 0.05 (Nilai P = 0.0174 <= 0.05).
- Berdasarkan Shapiro-Wilk Test, kolom "free sulfur dioxide" tidak dapat dianggap berdistribusi normal karena Nilai P-nya tidak lebih dari 0.05 (Nilai P = 0.0425 <= 0.05).

Kesimpulan

Berdasarkan tes normalitas yang dilakukan di atas, kolom "free sulfur dioxide" tidak berdistribusi normal.

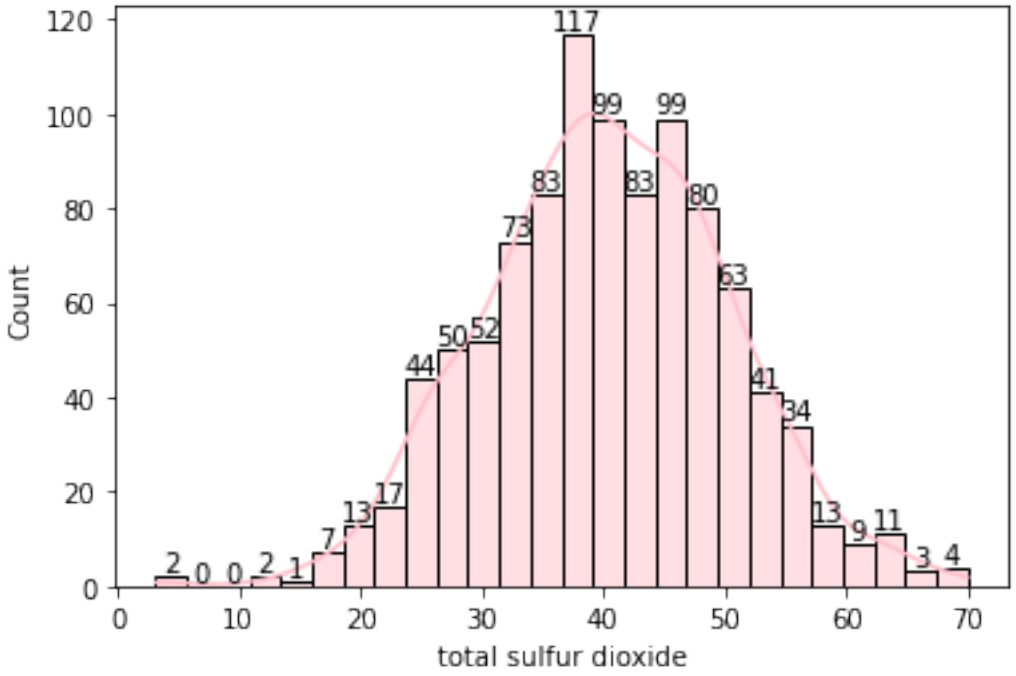
0.3.7 7. total sulfur dioxide

[9]:
```
normalityTests("total sulfur dioxide")
```

```
Skewness        : -0.024060026812269975
Excess Kurtosis : 0.06394978916172311

D'Agustino-Pearson Test
Test            : 0.3276640291639825
Nilai p         : 0.8488846101395726

Shapiro-Wilk Test
Test            : 0.9984723925590515
Nilai p         : 0.5367269515991211
```



Hasil Tes

- Berdasarkan histogram di atas, kolom "total sulfur dioxide" dapat dianggap berdistribusi normal karena histogramnya berbentuk *bell-shaped* yang simetris. *Skewness*-nya juga berada di antara -0.5 dan 0.5 (*skewness* = -0.0241, -0.5 < -0.0241 < 0.5), yang menandakan bahwa histogram di atas simetris. *Excess kurtosis*-nya juga berada di antara -0.5 dan 0.5 (*excess kurtosis* = 0.0639, -0.5 < 0.0639 < 0.5), yang menandakan bahwa histogram di atas memiliki keruncingan distribusi normal.
- Berdasarkan D'Agustino-Pearson Test, kolom "total sulfur dioxide" dapat dianggap berdistribusi normal karena Nilai P-nya lebih dari 0.05 (Nilai P = 0.8489 > 0.05).

- Berdasarkan Shapiro-Wilk Test, kolom “total sulfur dioxide” dapat dianggap berdistribusi normal karena Nilai P-nya lebih dari 0.05 (Nilai P = 0.5367 > 0.05).

Kesimpulan
Berdasarkan tes normalitas yang dilakukan di atas, kolom “total sulfur dioxide” berdistribusi normal.

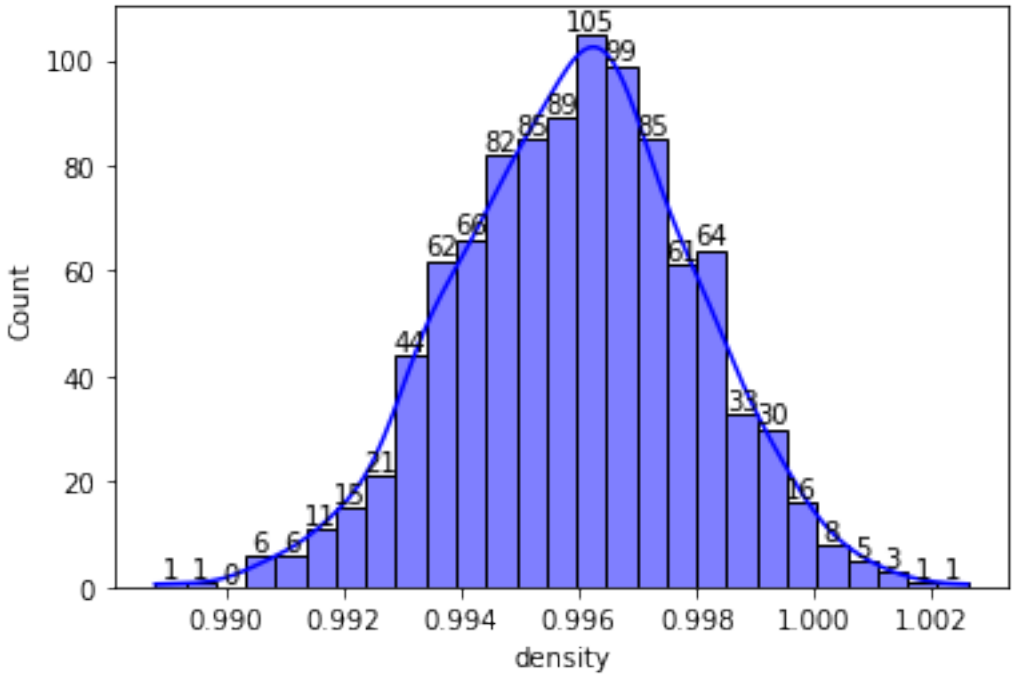
0.3.8 8. density

[10]:
```
normalityTests("density")
```

```
Skewness        : -0.07688278915513917
Excess Kurtosis : 0.01636562128503849

D'Agustino-Pearson Test
Test            : 1.026581544320803
Nilai p         : 0.5985227325531981

Shapiro-Wilk Test
Test            : 0.9989627003669739
Nilai p         : 0.8533204793930054
```



Hasil Tes

- Berdasarkan histogram di atas, kolom “density” dapat dianggap berdistribusi normal karena histogramnya berbentuk *bell-shaped* yang simetris. *Skewness*-nya juga berada di antara -0.5

dan 0.5 (*skewness* = -0.0769, -0.5 < -0.0769 < 0.5), yang menandakan bahwa histogram di atas simetris. *Excess kurtosis*-nya juga berada di antara -0.5 dan 0.5 (*excess kurtosis* = 0.0164, -0.5 < 0.0164 < 0.5), yang menandakan bahwa histogram di atas memiliki keruncingan distribusi normal.

- Berdasarkan D'Agustino-Pearson Test, kolom "density" dapat dianggap berdistribusi normal karena Nilai P-nya lebih dari 0.05 (Nilai P = 0.5985 > 0.05).
- Berdasarkan Shapiro-Wilk Test, kolom "density" dapat dianggap berdistribusi normal karena Nilai P-nya lebih dari 0.05 (Nilai P = 0.8533 > 0.05).

Kesimpulan
Berdasarkan tes normalitas yang dilakukan di atas, kolom "density" berdistribusi normal.

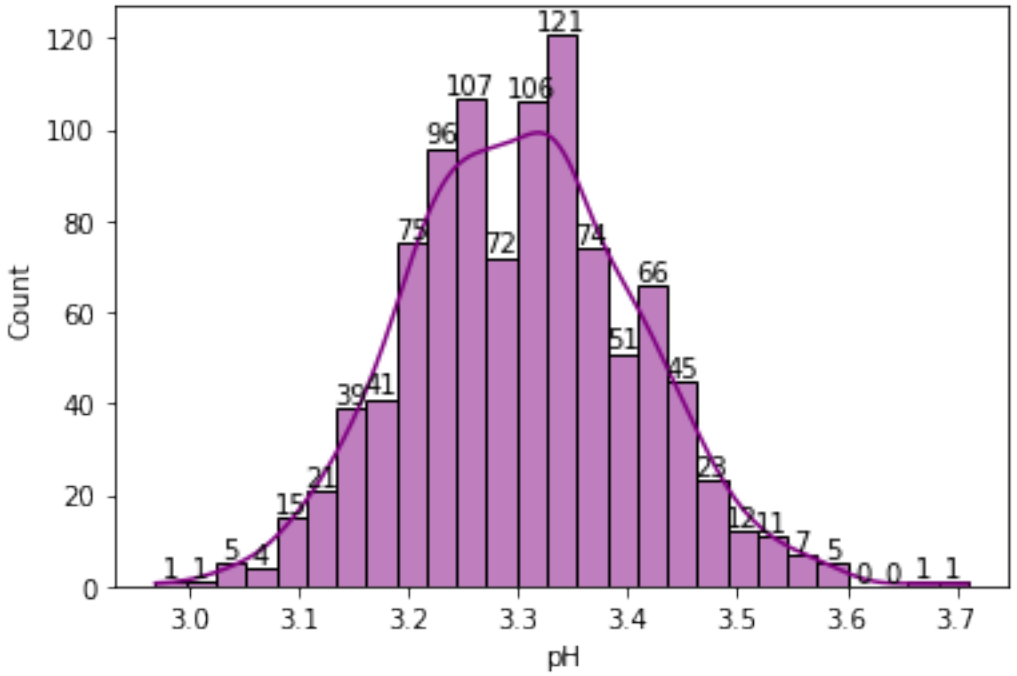
0.3.9 9. pH

[11]:
```
normalityTests("pH")
```

```
Skewness        : 0.14767259510827038
Excess Kurtosis : 0.0809095518741838

D'Agustino-Pearson Test
Test            : 3.9786546459928545
Nilai p         : 0.13678740824860436

Shapiro-Wilk Test
Test            : 0.997534453868866
Nilai p         : 0.13713516294956207
```



Hasil Tes

- Berdasarkan histogram di atas, kolom "pH" dapat dianggap berdistribusi normal karena histogramnya berbentuk *bell-shaped* yang simetris. *Skewness*-nya juga berada di antara -0.5 dan 0.5 (*skewness* = 0.1477, -0.5 < 0.1477 < 0.5), yang menandakan bahwa histogram di atas simetris. *Excess kurtosis*-nya juga berada di antara -0.5 dan 0.5 (*excess kurtosis* = 0.0809, -0.5 < 0.0809 < 0.5), yang menandakan bahwa histogram di atas memiliki keruncingan distribusi normal.
- Berdasarkan D'Agustino-Pearson Test, kolom "pH" dapat dianggap berdistribusi normal karena Nilai P-nya lebih dari 0.05 (Nilai P = 0.1368 > 0.05).
- Berdasarkan Shapiro-Wilk Test, kolom "pH" dapat dianggap berdistribusi normal karena Nilai P-nya lebih dari 0.05 (Nilai P = 0.1371 > 0.05).

Kesimpulan
Berdasarkan tes normalitas yang dilakukan di atas, kolom "pH" berdistribusi normal.

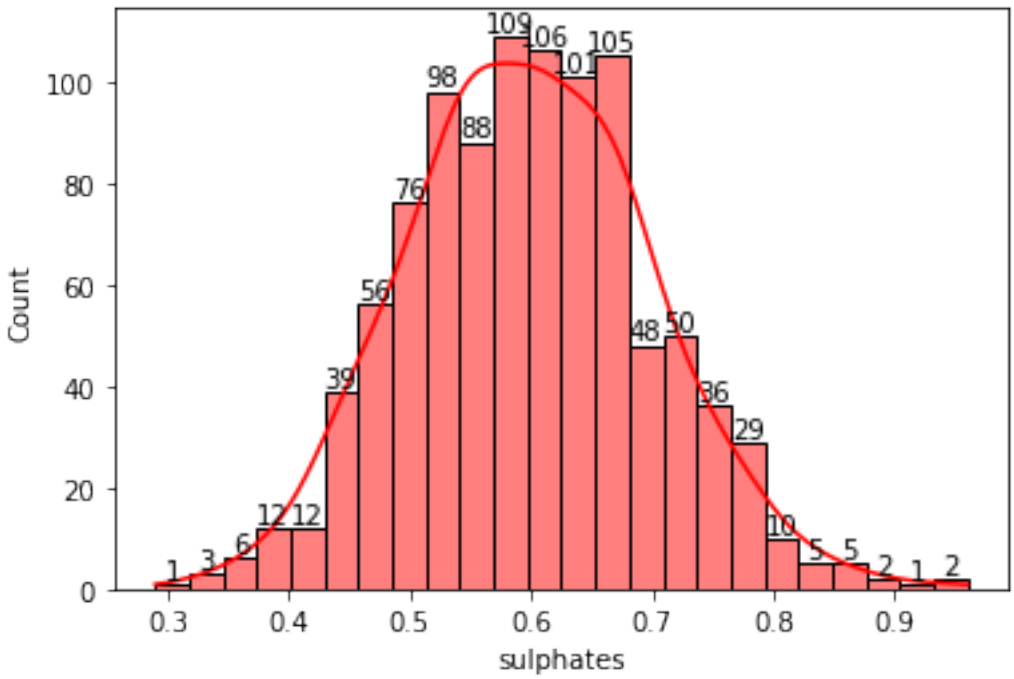
0.3.10 10. sulphates

[12]:
```
normalityTests("sulphates")
```

```
Skewness        : 0.1491989008699043
Excess Kurtosis : 0.06481928180859686

D'Agustino-Pearson Test
Test            : 3.948820277859041
Nilai p         : 0.13884318628391681

Shapiro-Wilk Test
Test            : 0.997409999370575
Nilai p         : 0.11214283108711243
```



Hasil Tes

- Berdasarkan histogram di atas, kolom "sulphates" dapat dianggap berdistribusi normal karena histogramnya berbentuk *bell-shaped* yang simetris. *Skewness*-nya juga berada di antara -0.5 dan 0.5 (*skewness* = 0.1492, -0.5 < 0.1492 < 0.5), yang menandakan bahwa histogram di atas simetris. *Excess kurtosis*-nya juga berada di antara -0.5 dan 0.5 (*excess kurtosis* = 0.0648, -0.5 < 0.0648 < 0.5), yang menandakan bahwa histogram di atas memiliki keruncingan distribusi normal.
- Berdasarkan D'Agustino-Pearson Test, kolom "sulphates" dapat dianggap berdistribusi normal karena Nilai P-nya lebih dari 0.05 (Nilai P = 0.1388 > 0.05).
- Berdasarkan Shapiro-Wilk Test, kolom "sulphates" dapat dianggap berdistribusi normal karena Nilai P-nya lebih dari 0.05 (Nilai P = 0.1121 > 0.05).

Kesimpulan
Berdasarkan tes normalitas yang dilakukan di atas, kolom "sulphates" berdistribusi normal.

0.3.11 11. alcohol

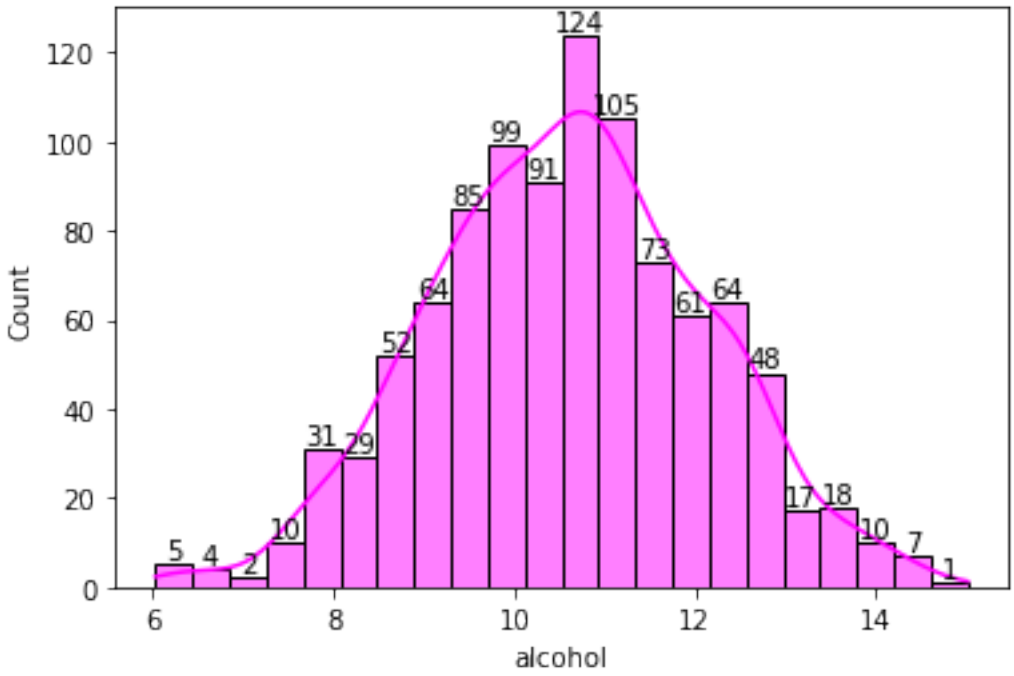
[13]:
```
normalityTests("alcohol")
```

```
Skewness        : -0.01899140432111647
Excess Kurtosis : -0.13173155932281988

D'Agustino-Pearson Test
Test            : 0.7740076714171271
```

```
Nilai p         : 0.6790884901361043

Shapiro-Wilk Test
Test            : 0.9984460473060608
Nilai p         : 0.519870400428772
```



Hasil Tes

- Berdasarkan histogram di atas, kolom "alcohol" dapat dianggap berdistribusi normal karena histogramnya berbentuk *bell-shaped* yang simetris. *Skewness*-nya juga berada di antara -0.5 dan 0.5 (*skewness* = -0.019, -0.5 < -0.019 < 0.5), yang menandakan bahwa histogram di atas simetris. *Excess kurtosis*-nya juga berada di antara -0.5 dan 0.5 (*excess kurtosis* = -0.1317, -0.5 < -0.1317 < 0.5), yang menandakan bahwa histogram di atas memiliki keruncingan distribusi normal.
- Berdasarkan D'Agustino-Pearson Test, kolom "alcohol" dapat dianggap berdistribusi normal karena Nilai P-nya lebih dari 0.05 (Nilai P = 0.6791 > 0.05).
- Berdasarkan Shapiro-Wilk Test, kolom "alcohol" dapat dianggap berdistribusi normal karena Nilai P-nya lebih dari 0.05 (Nilai P = 0.5199 > 0.05).

Kesimpulan
Berdasarkan tes normalitas yang dilakukan di atas, kolom "alcohol" berdistribusi normal.

0.3.12 12. quality

Kolom "quality" adalah kolom yang memiliki distribusi diskrit sehingga tidak perlu dilakukan pengecekan normalitas.

4

April 18, 2023

[7]:
```
import scipy.stats as st
import pandas as pd
import matplotlib.pyplot as plt
from statsmodels.stats.weightstats import ztest
from statsmodels.stats.proportion import proportions_ztest

dataAnggur = pd.read_csv('../data/anggur.csv')
```

Melakukan test hipotesis 1 sampel,

0.0.1 a. Nilai rata-rata pH di atas 3.29?

H_0 = Nilai rata-rata pH sama dengan 3.29 ($\mu = 3.29$)
H_1 = Nilai rata-rata pH lebih dari 3.29 ($\mu > 3.29$)

Tingkat Signifikan $\alpha = 0.05$

Lakukan uji statistik dengan one tailed test ke arah kanan (right tailed test) karena ($\mu > 3.29$). Ambil daerah kritis ($z > z_\alpha$)

Hitung nilai z dengan rumus

$$z = \frac{(x - \mu_0)}{(\sigma/\sqrt{n})}$$

Pengambilan Keputusan

Tes Daerah Kritis

- Reject H_0 jika ($z > z_\alpha$)
- Fail to reject H_0 jika ($z \leq z_\alpha$)

Tes Signifikansi

- Reject H_0 jika $p < \alpha$
- Fail to reject H_0 jika $p \geq \alpha$

[8]:
```
# Diketahui
rerata = 3.29
alpha = 0.05
```

```python
# Menggunakan ztest module, menghitung z dan p
z, p = ztest(dataAnggur['pH'], value = rerata)

# Menghitung z_alpha
z_a = st.norm.ppf(1-alpha)

# Hasil
print(f"Nilai z: {round(z, 4)}")
print(f"Nilai z_alpha: {round(z_a, 4)}")
print(f"Nilai p: {round(p, 6)}")
```

```
Nilai z: 4.1038
Nilai z_alpha: 1.6449
Nilai p: 4.1e-05
```

Hasil Tes

Tes Daerah Kritis

Karena z lebih besar dibandingkan dengan z_α $(4.103 > 1.644)$, reject H_0.

Tes Signifikansi

Karena p lebih kecil dibandingkan α $(0.000041 < 0.05)$, reject H_0.

Kesimpulan

Dengan tingkat signifikansi sebesar 0.05, ada bukti yang cukup untuk menolak klaim bahwa nilai rata-rata pH adalah 3.29. Maka nilai rata-rata pH lebih dari 3.29

0.0.2 b. Nilai rata-rata Residual Sugar tidak sama dengan 2.50?

H_0 = Nilai rata-rata Residual Sugar sama dengan 2.50 $(\mu = 2.50)$
H_1 = Nilai rata-rata Residual Sugar lebih dari 2.50 $(\mu \neq 2.50)$

Tingkat Signifikan $\alpha = 0.05$

Lakukan uji statistik dengan two tailed test pada bagian kanan $\mu > 2.50$ dengan $(z > z_{\alpha/2})$ dan bagian kiri $\mu < 2.50$ dengan $(z < -z_{\alpha/2})$

Hitung nilai z dengan rumus

$$z = \frac{(x - \mu_0)}{(\sigma/\sqrt{n})}$$

Pengambilan Keputusan

Tes Daerah Kritis

- Reject H_0 jika $z < -z_{\alpha/2}$ atau $z > z_{\alpha/2}$
- Fail to reject H_0 jika $-z_{\alpha/2} \leq z \leq z_{\alpha/2}$

Tes Signifikansi

- Reject H_0 jika $p < \alpha$
- Fail to reject H_0 jika $p \geq \alpha$

[9]:
```
# Diketahui
rerata = 2.50
alpha = 0.05

# Menggunakan ztest module, menghitung z dan p
z, p = ztest(dataAnggur['residual sugar'], value = rerata)

# Menghitung z_alpha
z_a = st.norm.ppf(1-(alpha/2))

# Hasil
print(f"Nilai z: {round(z, 4)}")
print(f"Nilai z_alpha/2: {round(z_a, 4)}")
print(f"Nilai p: {round(p, 6)}")
```

```
Nilai z: 2.148
Nilai z_alpha/2: 1.96
Nilai p: 0.031717
```

Hasil Tes
Tes Daerah Kritis
Karena z lebih besar dibandingkan dengan z_α $(2.148 > 1.96)$, reject H_0.
Tes Signifikansi
Karena p lebih kecil dibandingkan α $(0.031 < 0.05)$, reject H_0.

Kesimpulan
Dengan tingkat signifikansi sebesar 0.05, ada bukti yang cukup untuk menolak klaim bahwa nilai rata-rata residual sugar sama dengan 2.5. Maka nilai rata-rata Residual Sugar lebih dari 2.5.

0.0.3 c. Nilai rata-rata 150 baris pertama kolom sulphates bukan 0.65?

H_0 = Nilai rata-rata 150 baris pertama kolom sulphates sama dengan 0.65 $(\mu = 0.65)$
H_1 = Nilai rata-rata 150 baris pertama kolom sulphates tidak sama dengan 0.65 $(\mu \neq 0.65)$

Tingkat Signifikan $\alpha = 0.05$

Lakukan uji statistik dengan two tailed test pada bagian kanan $\mu > 0.65$ dengan $(z > z_{\alpha/2})$ dan bagian kiri $\mu < 0.65$ dengan $(z < -z_{\alpha/2})$

Hitung nilai z dengan rumus

$$z = \frac{(x - \mu_0)}{(\sigma/\sqrt{n})}$$

Pengambilan Keputusan

Tes Daerah Kritis

- Reject H_0 jika $z < -z_{\alpha/2}$ atau $z > z_{\alpha/2}$
- Fail to reject H_0 jika $-z_{\alpha/2} \leq z \leq z_{\alpha/2}$

Tes Signifikansi

- Reject H_0 jika $p < \alpha$
- Fail to reject H_0 jika $p \geq \alpha$

[10]:
```
# Diketahui
rerata = 0.65
alpha = 0.05

# Menggunakan ztest module, menghitung z dan p
z, p = ztest(dataAnggur['sulphates'].head(150), value = rerata)

# Menghitung z_alpha
z_a = st.norm.ppf(1-(alpha/2))

# Hasil
print(f"Nilai z: {round(z, 4)}")
print(f"Nilai z_alpha/2: {round(z_a, 4)}")
print(f"Nilai p: {round(p, 6)}")
```

```
Nilai z: -4.9648
Nilai z_alpha/2: 1.96
Nilai p: 1e-06
```

Hasil Tes
Tes Daerah Kritis
Karena z lebih kecil dibandingkan dengan $-z_{\alpha/2}$ ($-4.9648 < -1.96$), reject H_0.
Tes Signifikansi
Karena p lebih kecil dibandingkan α ($0.000001 < 0.05$), reject H_0.

Kesimpulan
Dengan tingkat signifikansi sebesar 0.05, ada bukti yang cukup untuk menolak klaim bahwa nilai rata-rata 150 baris pertama kolom sulphates sama dengan 0.65.

0.0.4 d. Nilai rata-rata total sulfur dioxide di bawah 35?

H_0 = Nilai rata-rata total sulfur dioxide sama dengan 35 ($\mu = 35$)
H_1 = Nilai rata-rata total sulfur dioxide kurang dari 35 ($\mu < 35$)

Tingkat Signifikan $\alpha = 0.05$

Lakukan uji statistik dengan one tailed test ke arah kiri (left tailed test) karena ($\mu < 35$). Ambil daerah kritis ($z < -z_{\alpha}$)

Hitung nilai z dengan rumus

$$z = \frac{(x - \mu_0)}{(\sigma/\sqrt{n})}$$

Pengambilan Keputusan

Tes Daerah Kritis

- Reject H_0 jika $(z < -z_\alpha)$
- Fail to reject H_0 jika $(z \geq z_\alpha)$

Tes Signifikansi

- Reject H_0 jika $p < \alpha$
- Fail to reject H_0 jika $p \geq \alpha$

[11]:
```
# Diketahui
rerata = 35
alpha = 0.05

# Menggunakan ztest module, menghitung z dan p
z, p = ztest(dataAnggur['total sulfur dioxide'], value = rerata)

# Menghitung z_alpha
z_a = st.norm.ppf(1-alpha)

# Hasil
print(f"Nilai z: {round(z, 4)}")
print(f"Nilai -z_alpha: -{round(z_a, 4)}")
# Karena merupakan one tailed test ke arah kiri, maka p-valuenya 1-p
print(f"Nilai p: {1-p}")
```

```
Nilai z: 16.7864
Nilai -z_alpha: -1.6449
Nilai p: 1.0
```

Hasil Tes
Tes Daerah Kritis
Karena z lebih besar dibandingkan dengan $-z_\alpha$ $(16.78 > -1.6449)$, fail to reject H_0.
Tes Signifikansi
Karena p lebih kecil dibandingkan α $(1 > 0.05)$, fail to reject H_0.

Kesimpulan
Dengan tingkat signifikansi sebesar 0.05, tidak ada bukti yang cukup untuk menolak klaim bahwa nilai rata-rata total sulfur dioxide sama dengan 35.

0.0.5 e. Proporsi nilai total Sulfat Dioxide yang lebih dari 40, adalah tidak sama dengan 50% ?

H_0 = Proporsi nilai Conductivity yang lebih dari 40 sama dengan 50 ($p = 50\%$)
H_1 = Proporsi nilai Conductivity yang lebih dari 40 tidak sama dengan 50 ($p \neq 50\%$)

Tingkat Signifikan $\alpha = 0.05$

Lakukan uji statistik dengan two tailed test pada bagian kanan dengan ($z > z_{\alpha/2}$) dan bagian kiri dengan ($z < -z_{\alpha/2}$)

Hitung nilai z dengan rumus

$$z = \frac{(p - p_0)}{(p_0 q_0 / \sqrt{n})}$$

Pengambilan Keputusan

Tes Daerah Kritis

- Reject H_0 jika $z < -z_{\alpha/2}$ atau $z > z_{\alpha/2}$
- Fail to reject H_0 jika $-z_{\alpha/2} \leq z \leq z_{\alpha/2}$

Tes Signifikansi

- Reject H_0 jika $p < \alpha$
- Fail to reject H_0 jika $p \geq \alpha$

[12]:
```
p_0 = 0.50
alpha = 0.05

# Test
TotalSD_over_40 = len(dataAnggur[dataAnggur["total sulfur dioxide"] > 40])

z, p = proportions_ztest(TotalSD_over_40,
                         len(dataAnggur), value=p_0, prop_var=p_0)
z_a = st.norm.ppf(1-alpha/2)

# Results
print(f"Nilai z: {round(z, 4)}")
print(f"Nilai z_alpha/2: {round(z_a, 4)}")
print(f"Nilai p: {round(p, 4)}")
```

```
Nilai z: 0.7589
Nilai z_alpha/2: 1.96
Nilai p: 0.4479
```

Hasil Tes
Tes Daerah Kritis
Karena z memenuhi $-z_\alpha < z < z_\alpha$ ($-1.96 < 0.758 < 1.96$), fail to reject H_0.

Tes Signifikansi
Karena p lebih besar dibandingkan α ($0.447 > 0.05$), fail to reject H_0.

Kesimpulan
Dengan tingkat signifikansi sebesar 0.05, tidak ada bukti yang cukup untuk menolak klaim bahwa proporsi nilai sulfat dioxide yang lebih dari 40 adalah sama dengan 50%.

5

April 18, 2023

[26]:
```
import pandas as pd
import matplotlib.pyplot as plt
import scipy.stats as st
from statsmodels.stats.weightstats import ztest
from statsmodels.stats.proportion import proportions_ztest

dataAnggur = pd.read_csv('../data/anggur.csv')
```

Melakukan test hipotesis 2 sampel,

0.0.1 a. Data kolom fixed acidity dibagi 2 sama rata: bagian awal dan bagian akhir kolom. Benarkah rata-rata kedua bagian tersebut sama?

Hipotesis
Misalkan 1 melambangkan bagian awal kolom fixed acidity dan 2 melambangkan bagian akhir kolom fixed acidity.
H_0 : Rata-rata kedua bagian sama ($\mu_1 = \mu_2, \mu_1 - \mu_2 = 0$)
H_1 : Rata-rata kedua bagian berbeda ($\mu_1 \neq \mu_2, \mu_1 - \mu_2 \neq 0$)
Dari kalimat soal, kita dapat menganggap bahwa klaimnya adalah H_0.

Tingkat Signifikansi
$\alpha = 0.05$

Uji Statistik
Pada pengujian hipotesis ini, meskipun variansi populasi tidak diketahui, digunakan z-test, bukan t-test. Hal ini diputuskan karena jumlah sampel yang digunakan jauh lebih banyak dibanding 30.

Digunakan tes statistik z dengan rumus:

$$z = \frac{(\bar{x}_1 - \bar{x}_2) - (\mu_1 - \mu_2)}{\sqrt{\sigma_1^2/n_1 + \sigma_2^2/n_2}}$$

dengan daerah kritis:
$z < -z_{\alpha/2}$ atau $z > z_{\alpha/2}$ (two-tailed test)

Pengambilan Keputusan

Tes Daerah Kritis

- Reject H_0 jika $z < -z_{\alpha/2}$ atau $z > z_{\alpha/2}$
- Fail to reject H_0 jika $-z_{\alpha/2} \leq z \leq z_{\alpha/2}$

Tes Signifikansi

- Reject H_0 jika $p < \alpha$
- Fail to reject H_0 jika $p \geq \alpha$

[27]:
```
# Diketahui
alpha = 0.05
deltaMean = 0

# Ambil data
nData = len(dataAnggur) // 2
dataAwal = dataAnggur["fixed acidity"][ : nData]
dataAkhir = dataAnggur["fixed acidity"][nData : ]

# Lakukan z-test dengan memanfaatkan library statsmodels untuk mendapatkan␣
↪nilai z dan p
z, p = ztest(dataAwal, dataAkhir, value = deltaMean)

# Hitung z_alpha/2
zAlpha2 = st.norm.ppf(1 - alpha / 2)

# Tampilkan hasil
print(f"Nilai z               : {round(z, 5)}")
print(f"Nilai z_alpha/2       : {round(zAlpha2, 5)}")
print(f"Nilai p               : {round(p, 5)}")
```

```
Nilai z               : 0.02604
Nilai z_alpha/2       : 1.95996
Nilai p               : 0.97922
```

Hasil Tes

Tes Daerah Kritis
Karena $-z_{\alpha/2} \leq z \leq z_{\alpha/2}$ $(-1.95996 \leq 0.02604 \leq 1.95996)$, fail to reject H_0.

Tes Signifikansi
Karena $p \geq \alpha$ $(0.97922 \geq 0.05)$, fail to reject H_0.

Kesimpulan
Dengan tingkat signifikansi sebesar 0.05, tidak ada bukti yang cukup untuk menolak klaim bahwa rerata bagian awal dan akhir kolom fixed acidity bernilai sama.

0.0.2 b. Data kolom chlorides dibagi 2 sama rata: bagian awal dan bagian akhir kolom. Benarkah rata-rata bagian awal lebih besar daripada bagian akhir sebesar 0.001?

Hipotesis
Misalkan 1 melambangkan bagian awal kolom chlorides dan 2 melambangkan bagian akhir kolom chlorides.
H_0 : Rata-rata bagian awal lebih besar daripada bagian akhir sebesar 0.001 ($\mu_1 = \mu_2 + 0.001, \mu_1 - \mu_2 = 0.001$)
H_1 : Rata-rata bagian awal tidak lebih besar daripada bagian akhir sebesar 0.001 ($\mu_1 \neq \mu_2 + 0.001, \mu_1 - \mu_2 \neq 0.001$)
Dari kalimat soal, kita dapat menganggap bahwa klaimnya adalah H_0.

Tingkat Signifikansi
$\alpha = 0.05$

Uji Statistik
Pada pengujian hipotesis ini, meskipun variansi populasi tidak diketahui, digunakan z-test, bukan t-test. Hal ini diputuskan karena jumlah sampel yang digunakan jauh lebih banyak dibanding 30.

Digunakan tes statistik z dengan rumus:

$$z = \frac{(\bar{x}_1 - \bar{x}_2) - (\mu_1 - \mu_2)}{\sqrt{\sigma_1^2/n_1 + \sigma_2^2/n_2}}$$

dengan daerah kritis:
$z < -z_{\alpha/2}$ atau $z > z_{\alpha/2}$ (two-tailed test)

Pengambilan Keputusan

Tes Daerah Kritis

- Reject H_0 jika $z < -z_{\alpha/2}$ atau $z > z_{\alpha/2}$
- Fail to reject H_0 jika $-z_{\alpha/2} \leq z \leq z_{\alpha/2}$

Tes Signifikansi

- Reject H_0 jika $p < \alpha$
- Fail to reject H_0 jika $p \geq \alpha$

[28]:
```
# Diketahui
alpha = 0.05
deltaMean = 0.001

# Ambil data
nData = len(dataAnggur) // 2
dataAwal = dataAnggur["chlorides"][ : nData]
dataAkhir = dataAnggur["chlorides"][nData : ]
```

```
# Lakukan z-test dengan memanfaatkan library statsmodels untuk mendapatkan␣
 ↪nilai z dan p
z, p = ztest(dataAwal, dataAkhir, value = deltaMean)

# Hitung z_alpha/2
zAlpha2 = st.norm.ppf(1 - alpha / 2)

# Tampilkan hasil
print(f"Nilai z               : {round(z, 5)}")
print(f"Nilai z_alpha/2       : {round(zAlpha2, 5)}")
print(f"Nilai p               : {round(p, 5)}")
```

```
Nilai z               : -0.46732
Nilai z_alpha/2       : 1.95996
Nilai p               : 0.64027
```

Hasil Tes

Tes Daerah Kritis
Karena $-z_{\alpha/2} \leq z \leq z_{\alpha/2}$ $(-1.95996 \leq -0.46732 \leq 1.95996)$, fail to reject H_0.

Tes Signifikansi
Karena $p \geq \alpha$ $(0.64027 \geq 0.05)$, fail to reject H_0.

Kesimpulan
Dengan tingkat signifikansi sebesar 0.05, tidak ada bukti yang cukup untuk menolak klaim bahwa untuk kolom chlorides, rata-rata bagian awal lebih besar daripada bagian akhir sebesar 0.001.

0.0.3 c. Benarkah rata-rata sampel 25 baris pertama kolom Volatile Acidity sama dengan rata-rata 25 baris pertama kolom Sulphates?

Hipotesis
Misalkan 1 melambangkan 25 baris pertama kolom volatile acidity dan 2 melambangkan 25 baris pertama kolom sulphates.
H_0 : Rata-rata 25 baris pertama kolom volatile acidity sama dengan rata-rata 25 baris pertama kolom sulphates $(\mu_1 = \mu_2, \mu_1 - \mu_2 = 0)$
H_1 : Rata-rata 25 baris pertama kolom volatile acidity tidak sama dengan rata-rata 25 baris pertama kolom sulphates $(\mu_1 \neq \mu_2, \mu_1 - \mu_2 \neq 0)$
Dari kalimat soal, kita dapat menganggap bahwa klaimnya adalah H_0.

Tingkat Signifikansi
$\alpha = 0.05$

Uji Statistik
Pada pengujian hipotesis ini, karena variansi populasi tidak diketahui dan banyak sampel kurang

dari 30, digunakan t-test. Dipilih kasus untuk variansi populasi yang berbeda karena diasumsikan kedua data yang berbeda kolom memiliki variansi populasi yang berbeda.

Digunakan tes statistik t dengan rumus:

$$t = \frac{(\bar{x}_1 - \bar{x}_2) - (\mu_1 - \mu_2)}{\sqrt{s_1^2/n_1 + s_2^2/n_2}}$$

dengan derajat kebebasan:

$$v = \frac{(s_1^2/n_1 + s_2^2/n_2)^2}{\frac{(s_1^2/n_1)^2}{n_1 - 1} + \frac{(s_2^2/n_2)^2}{n_2 - 1}}$$

dengan daerah kritis:
$t < -t_{\alpha/2}$ atau $t > t_{\alpha/2}$ (two-tailed test)

Pengambilan Keputusan

Tes Daerah Kritis

- Reject H_0 jika $t < -t_{\alpha/2}$ atau $t > t_{\alpha/2}$
- Fail to reject H_0 jika $-t_{\alpha/2} \leq t \leq t_{\alpha/2}$

Tes Signifikansi

- Reject H_0 jika $p < \alpha$
- Fail to reject H_0 jika $p \geq \alpha$

[29]:
```
# Diketahui
alpha = 0.05
deltaMean = 0

# Ambil data
nData = 25
dataVolatileAcidity = dataAnggur["volatile acidity"][ : nData]
dataSulphates = dataAnggur["sulphates"][ : nData]

# Lakukan t-test dengan memanfaatkan library scipy untuk mendapatkan nilai t␣
↪dan p
t, p = st.ttest_ind(a=dataVolatileAcidity, b=dataSulphates, equal_var=False)

# Hitung derajat kebebasan
s1_2 = dataVolatileAcidity.var()
s2_2 = dataSulphates.var()
n1 = len(dataVolatileAcidity)
n2 = len(dataSulphates)
v = (s1_2/n1 + s2_2/n2)**2 / (((s1_2/n1)**2)/(n1-1) + ((s2_2/n2)**2)/(n2-1))

# Hitung t_alpha/2
tAlpha2 = st.t.ppf(q=1-alpha/2,df=v)
```

```
# Tampilkan hasil
print(f"Nilai t                : {round(t, 5)}")
print(f"Nilai t_alpha/2        : {round(tAlpha2, 5)}")
print(f"Nilai p                : {round(p, 5)}")
```

```
Nilai t                : -2.63748
Nilai t_alpha/2        : 2.01593
Nilai p                : 0.01153
```

Hasil Tes

Tes Daerah Kritis
Karena $t < -t_{\alpha/2}$ $(-2.63748 < -2.01593)$, reject H_0.

Tes Signifikansi
Karena $p < \alpha$ $(0.01153 < 0.05)$, reject H_0.

Kesimpulan
Dengan tingkat signifikansi sebesar 0.05, ada bukti yang cukup untuk menolak klaim bahwa rata-rata 25 baris pertama kolom volatile acidity sama dengan rata-rata 25 baris pertama kolom sulphates.

0.0.4 d. Bagian awal kolom residual sugar memiliki variansi yang sama dengan bagian akhirnya?

Hipotesis
Misalkan 1 melambangkan bagian awal kolom residual sugar dan 2 melambangkan bagian akhir kolom residual sugar.
H_0 : Variansi bagian awal kolom residual sugar sama dengan bagian akhirnya $(\sigma_1^2 = \sigma_2^2)$
H_1 : Variansi bagian awal kolom residual sugar tidak sama dengan bagian akhirnya $(\sigma_1^2 \neq \sigma_2^2)$
Dari kalimat soal, kita dapat menganggap bahwa klaimnya adalah H_0.

Tingkat Signifikansi
$\alpha = 0.05$

Uji Statistik
Pada uji hipotesis ini, digunakan tes statistik f dengan rumus:

$$f = \frac{s_1^2}{s_1^2}$$

dengan daerah kritis:
$f < f_{\alpha/2}(v_1, v_2)$ atau $f > f_{1-\alpha/2}(v_1, v_2)$ (two-tailed test) dengan $v_1 = n_1 - 1$ dan $v_2 = n_2 - 1$

Pengambilan Keputusan

Tes Daerah Kritis

- Reject H_0 jika $f < f_{\alpha/2}(v_1, v_2)$ atau $f > f_{1-\alpha/2}(v_1, v_2)$
- Fail to reject H_0 jika $f_{\alpha/2}(v_1, v_2) \leq f \leq f_{1-\alpha/2}(v_1, v_2)$

Tes Signifikansi

- Reject H_0 jika $p < \alpha$
- Fail to reject H_0 jika $p \geq \alpha$

[30]:
```
# Diketahui
alpha = 0.05

# Ambil data
nData = len(dataAnggur) // 2
dataAwal = dataAnggur["residual sugar"][ : nData]
dataAkhir = dataAnggur["residual sugar"][ nData : ]

# Hitung nilai f
f = dataAwal.var() / dataAkhir.var()

# Tentukan derajat kebebasan
v1 = len(dataAwal) - 1
v2 = len(dataAkhir) - 1

# Hitung f_(1 - alpha/2) dan f_alpha/2 dengan library scipy
f1MinAlpha2 = st.f.ppf(1 - alpha/2, v1, v2)
fAlpha2 = st.f.ppf(alpha/2, v1, v2)

# Hitung nilai p, p untuk two-tailed test adalah 2 kali tail area
p = st.f.cdf(f, v1, v2) * 2

# Tampilkan hasil
print(f"Nilai f               : {round(f, 5)}")
print(f"Nilai f_(1 - alpha/2) : {round(f1MinAlpha2, 5)}")
print(f"Nilai f_alpha/2       : {round(fAlpha2, 5)}")
print(f"Nilai p               : {round(p, 5)}")
```

```
Nilai f               : 0.942
Nilai f_(1 - alpha/2) : 1.19206
Nilai f_alpha/2       : 0.83889
Nilai p               : 0.50482
```

Hasil Tes

Tes Daerah Kritis

Karena $f_{\alpha/2}(v_1, v_2) \leq f \leq f_{1-\alpha/2}(v_1, v_2)$ $(0.83889 \leq 0.942 \leq 1.19206)$, fail to reject H_0.

Tes Signifikansi
Karena $p \geq \alpha$ ($0.74759 \geq 0.05$), fail to reject H_0.

Kesimpulan
Dengan tingkat signifikansi sebesar 0.05, tidak ada bukti yang cukup untuk menolak klaim bahwa variansi bagian awal dan akhir kolom residual sugar bernilai sama.

0.0.5 e. Proporsi nilai setengah bagian awal alcohol yang lebih dari 7, adalah lebih besar daripada, proporsi nilai yang sama di setengah bagian akhir alcohol?

Hipotesis
Misalkan 1 melambangkan setengah bagian awal kolom alcohol yang lebih dari 7 dan 2 melambangkan setengah bagian akhir kolom kolom alcohol yang lebih dari 7.
H_0 : Proporsi nilai setengah bagian awal alcohol yang lebih dari 7 sama dengan proporsi nilai yang sama di setengah bagian akhir alcohol ($p_1 = p_2, p_1 - p_2 = 0$)
H_1 : Proporsi nilai setengah bagian awal alcohol yang lebih dari 7 lebih besar daripada proporsi nilai yang sama di setengah bagian akhir alcohol ($p_1 > p_2, p_1 - p_2 > 0$)
Dari kalimat soal, kita dapat menganggap bahwa klaimnya adalah H_1.

Tingkat Signifikansi
$\alpha = 0.05$

Uji Statistik
Pada uji hipotesis ini, digunakan tes statistik z dengan rumus:

$$z = \frac{\hat{p_1} - \hat{p_2}}{\sqrt{\hat{p}\hat{q}(1/n_1 + 1/n_2)}}$$

dengan $\hat{p}$:

$$\hat{p} = \frac{x_1 + x_2}{n_1 + n_2}, \hat{q} = 1 - \hat{p}$$

dengan daerah kritis:
$z > z_{\alpha/2}$ (one-tailed test)

Pengambilan Keputusan

Tes Daerah Kritis

- Reject H_0 jika $z > z_\alpha$
- Fail to reject H_0 jika $z \leq z_\alpha$

Tes Signifikansi

- Reject H_0 jika $p < \alpha$
- Fail to reject H_0 jika $p \geq \alpha$

[31]:
```
# Diketahui
alpha = 0.05
```

```
deltaProp = 0

# Ambil data
nData = len(dataAnggur) // 2
dataAwal = dataAnggur[ : nData ]
dataAkhir = dataAnggur[ nData : ]

# Lakukan proportions z-test dengan memanfaatkan library statsmodels untuk␣
 ↪mendapatkan nilai z dan p
xAwal = len(dataAwal[dataAwal["alcohol"] > 7])
nAwal = len(dataAwal)
xAkhir = len(dataAkhir[dataAkhir["alcohol"] > 7])
nAkhir = len(dataAkhir)
z, _ = proportions_ztest([xAwal, xAkhir], [nAwal, nAkhir], value = deltaProp,␣
 ↪prop_var = deltaProp)
p = 1 - st.norm.cdf(z)

# Hitung z_alpha
zAlpha = st.norm.ppf(1 - alpha)

# Tampilkan hasil
print(f"Nilai z               : {round(z, 5)}")
print(f"Nilai z_alpha         : {round(zAlpha, 5)}")
print(f"Nilai p               : {round(p, 5)}")
```

```
Nilai z               : 0.0
Nilai z_alpha         : 1.64485
Nilai p               : 0.5
```

Hasil Tes

Tes Daerah Kritis
Karena $z \leq z_\alpha$ $(0.0 \leq 1.64485)$, fail to reject H_0.

Tes Signifikansi
Karena $p \geq \alpha$ $(1.0 \geq 0.05)$, fail to reject H_0.

Kesimpulan
Dengan tingkat signifikansi sebesar 0.05, tidak ada bukti yang cukup untuk mendukung klaim bahwa proporsi nilai setengah bagian awal alcohol yang lebih dari 7 lebih besar daripada proporsi nilai yang sama di setengah bagian akhir alcohol.