accident-par-regression-logistique

September 30, 2025

1 Introduction

Chaque année, le nombre d'accidents de la route continue de constituer une préoccupation majeure en matière de santé publique et de sécurité routière. L'impact de ces événements dépasse largement les seuls dommages humains, engendrant aussi des coûts économiques et sociaux importants. La complexité des causes d'accidents repose sur l'interaction de multiples facteurs liés aux conditions météorologiques, aux caractéristiques des routes, au comportement des conducteurs, à la densité du trafic, ainsi qu'au type de véhicule. Ces éléments, pris isolément ou combinés, influencent la probabilité qu'un accident survienne.

Cette étude statistique vise à explorer et quantifier l'impact de ces facteurs sur la survenue d'accidents routiers à partir d'un ensemble de données comprenant 840 observations détaillées. La problématique principale posée est la suivante : Quels sont les facteurs significativement associés à la survenue d'un accident et comment peut-on modéliser le risque d'accident en fonction de ces facteurs pour améliorer les stratégies de prévention ?

[25]:
```
import pandas as pd
import numpy as np
from scipy.stats.mstats import winsorize
from scipy.stats import chi2_contingency, fisher_exact
from scipy.stats import f_oneway, kruskal, shapiro, levene, bartlett
from scipy.stats import chi2
from sklearn.metrics import roc_curve
import statsmodels.api as sm
import math
import seaborn as sns
import matplotlib.pyplot as plt
import warnings
warnings.filterwarnings("ignore")
```

2 I- Exploration des données

2.1 Importation des données

- Visualisation des 5 premieres lignes

[26]:
```
df = pd.read_excel("dataset_accident.xlsx")
df.head()
```

[26]:

	Meteo	Type_de_route	Heure_du_jour	Densité_du_trafic	\
0	Pluvieux	Route urbaine	Matin	Densite moderee	
1	Clair	Route rurale	Nuit	NaN	
2	Pluvieux	Autoroute	Soir	Densite moderee	
3	Clair	Route urbaine	Après-midi	Forte densite	
4	Pluvieux	Autoroute	Matin	Densite moderee	

	Limitation_de_vitesse	Nombre_de_véhicules	Conducteur_Alcool	\
0	100.0	5.0	Non	
1	120.0	3.0	Non	
2	60.0	4.0	Non	
3	60.0	3.0	Non	
4	195.0	11.0	Non	

	Gravité_de_l'accident	Etat_de_la_route	Type_de_vehicule	Age_du_conducteur	\
0	NaN	Humide	Voiture	51.0	
1	Moderee	Humide	Camion	49.0	
2	Faible	Glace	Voiture	54.0	
3	Faible	En construction	Bus	34.0	
4	Faible	Sec	Voiture	62.0	

	Experience_du_conducteur	condition_eclairage_route	Accident
0	48.0	Lumiere artificielle	Non
1	43.0	Lumiere artificielle	Non
2	52.0	Lumiere artificielle	Non
3	31.0	Lumiere du jour	Non
4	55.0	Lumiere artificielle	Oui

2.2 Informations sur la donnée

Météo : L'impact des conditions météorologiques sur la probabilité d'accidents - Clair : Aucune condition météorologique défavorable. - Pluvieux : Les conditions pluvieuses augmentent les risques d'accidents. - Brouillard : Les conditions de brouillard réduisent la visibilité, augmentant ainsi les risques d'accident. - Neigeux : La neige peut rendre les routes glissantes et augmenter la probabilité d'accident. - Orageux : Un temps orageux peut créer des conditions de conduite dangereuses.

Road_Type : Le type de route, influençant la probabilité d'accidents. - Autoroute : routes à grande vitesse avec des risques plus élevés d'accidents graves. - Route urbaine : routes situées dans les limites de la ville, généralement avec plus de trafic et des vitesses plus faibles. - Route rurale : Routes situées en dehors des zones urbaines, souvent avec moins de véhicules et des vitesses plus faibles. - Route de montagne : Routes avec courbes et changements d'altitude, augmentant le risque d'accident.

Time_of_Day : L'heure de la journée à laquelle l'accident se produit. - Matin : La période entre le lever du soleil et midi. - Après-midi : La période entre midi et le soir. - Soir : La période juste avant le coucher du soleil. - Nuit : La nuit, souvent associée à une visibilité réduite et à un risque plus élevé.

Traffic_Density : Le niveau de trafic sur la route. - 0 : Faible densité (peu de véhicules). - 1 :

Densité modérée. - 2 : Forte densité (nombreux véhicules).

Speed_Limit : La vitesse maximale autorisée sur la route.

Number_of_Vehicles : Le nombre de véhicules impliqués dans l'accident, allant de 1 à 5.

Driver_Alcohol : Si le conducteur a consommé de l'alcool. - 0 : Aucune consommation d'alcool.(Non) - 1 : La consommation d'alcool (qui augmente le risque d'accident).(Oui)

Accident_Severity : La gravité de l'accident. - Faible : Accident mineur. - Modéré : Accident modéré avec quelques dégâts ou blessures. - Élevé : Accident grave avec dommages ou blessures importants.

Road_Condition : L'état de la surface de la route. - Sec : Routes sèches avec un risque minimal. - Humide : Routes mouillées à cause de la pluie, augmentant le risque d'accidents. - Glacé : Glace sur la route, augmentant considérablement le risque d'accident. - En construction : Routes en construction, qui peuvent présenter des obstacles ou une mauvaise qualité de route.

Vehicle_Type : Le type de véhicule impliqué dans l'accident. - Voiture : Une voiture de tourisme ordinaire. - Camion : Un gros véhicule utilisé pour transporter des marchandises. - Motocyclette : Véhicule motorisé à deux roues. - Bus : Un grand véhicule utilisé pour le transport public.

Driver_Age : Âge du conducteur. Les valeurs varient de 18 à 70 ans.

Expérience_Conducteur : années d'expérience du conducteur. Les valeurs varient de 0 à 50 ans.

Road_Light_Condition : Les conditions d'éclairage sur la route. - Lumière du jour : pendant la journée, lorsque la visibilité est généralement bonne. - Lumière artificielle : La route est éclairée par des lampadaires. - Pas de lumière : la route n'est pas éclairée, généralement pendant la nuit dans les zones mal éclairées.

[27]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 840 entries, 0 to 839
Data columns (total 14 columns):
 #   Column                     Non-Null Count  Dtype
---  ------                     --------------  -----
 0   Meteo                      798 non-null    object
 1   Type_de_route              798 non-null    object
 2   Heure_du_jour              798 non-null    object
 3   Densité_du_trafic          798 non-null    object
 4   Limitation_de_vitesse      798 non-null    float64
 5   Nombre_de_véhicules        798 non-null    float64
 6   Conducteur_Alcool          798 non-null    object
 7   Gravité_de_l'accident      798 non-null    object
 8   Etat_de_la_route           798 non-null    object
 9   Type_de_vehicule           798 non-null    object
 10  Age_du_conducteur          798 non-null    float64
 11  Experience_du_conducteur   798 non-null    float64
 12  condition_eclairage_route  798 non-null    object
 13  Accident                   798 non-null    object
```

```
dtypes: float64(4), object(10)
memory usage: 92.0+ KB
```

- **Les données contiennent 840 lignes et 14 variables dont 4 variables quantitatives et 10 qualitatives.**
- **Dans chaque colonne des variables, sur les 840 lignes seuls 798 contiennent des valeurs non nulles.**

2.3 Traitement de la donnée

- Valeurs manquantes

[28]:
```
df.isnull().sum()
```

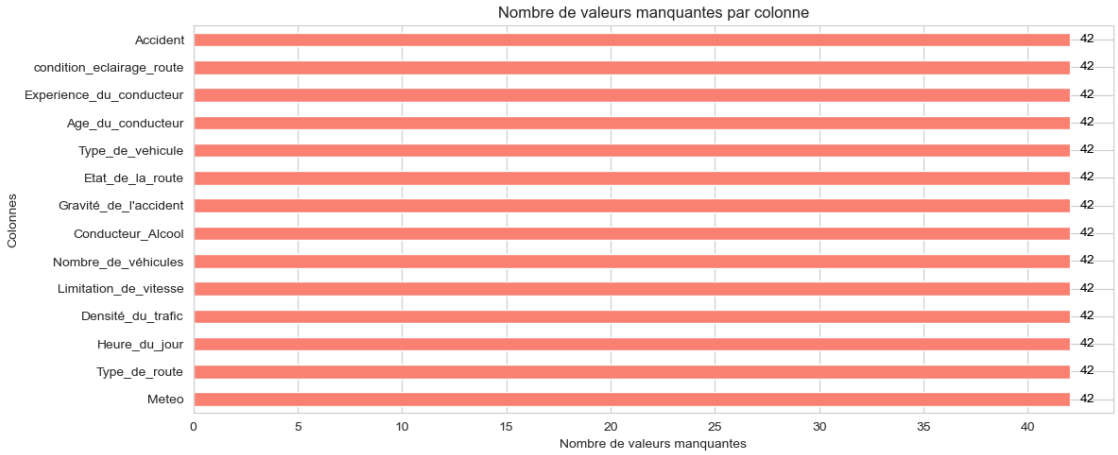
[28]:
```
Meteo                        42
Type_de_route                42
Heure_du_jour                42
Densité_du_trafic            42
Limitation_de_vitesse        42
Nombre_de_véhicules          42
Conducteur_Alcool            42
Gravité_de_l'accident        42
Etat_de_la_route             42
Type_de_vehicule             42
Age_du_conducteur            42
Experience_du_conducteur     42
condition_eclairage_route    42
Accident                     42
dtype: int64
```

[29]:
```
pourcentage_total = (df.isnull().sum().sum() / df.size) * 100
print(f"Pourcentage global de valeurs manquantes : {pourcentage_total:.1f}%")
```

```
Pourcentage global de valeurs manquantes : 5.0%
```

[30]:
```
missing_counts = df.isnull().sum()
missing_counts = missing_counts[missing_counts > 0]
missing_counts_sorted = missing_counts.sort_values()

plt.figure(figsize=(12, 5))
ax = missing_counts_sorted.plot(kind='barh', color='salmon')
plt.title("Nombre de valeurs manquantes par colonne")
plt.xlabel("Nombre de valeurs manquantes")
plt.ylabel("Colonnes")

for i, v in enumerate(missing_counts_sorted):
    ax.text(v + 0.5, i, str(int(v)), color='black', va='center')

plt.tight_layout()
```

```
plt.show()
```



On remarque que chaque variable comporte 42 valeurs manquantes, ce qui nécessite une prise en charge attentive. Avant de choisir une méthode d'imputation, il est essentiel de comparer les statistiques descriptives avant et après le traitement. Cela permet d'évaluer l'impact de chaque technique sur la distribution des données. L'objectif est de sélectionner la méthode qui altère le moins les caractéristiques initiales du jeu de données, afin de préserver sa fiabilité et éviter tout biais dans les analyses futures.

[31]:
```
print("TRAITEMENT VARIABLE QUANTI FINI   ")
df = df.fillna(df.median(numeric_only=True))
```

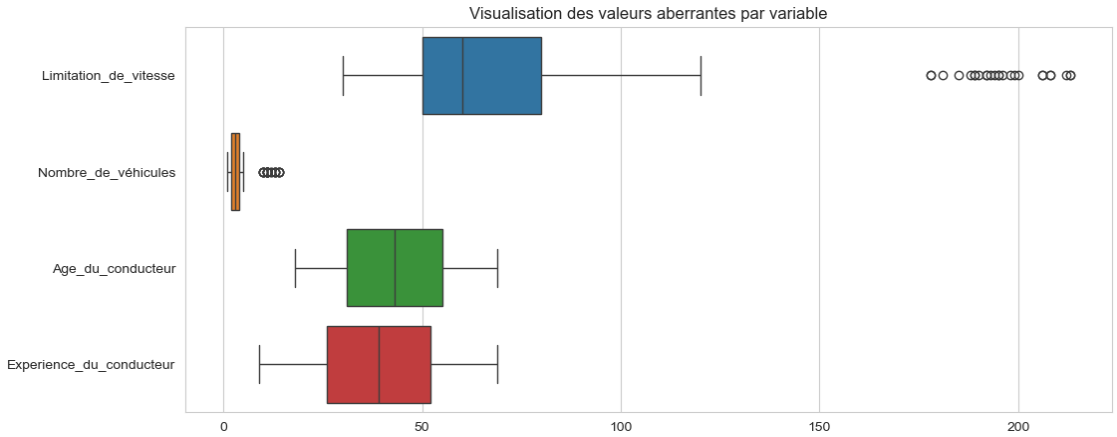
```
TRAITEMENT VARIABLE QUANTI FINI
```

[32]:
```
print("TRAITEMENT VARIABLE QUALI FINI   ")
for col in df.select_dtypes(include='object'):
    mode = df[col].mode()[0]
    df[col].fillna(mode, inplace=True)
```

```
TRAITEMENT VARIABLE QUALI FINI
```

- Valeurs aberrantes – Avant traitement

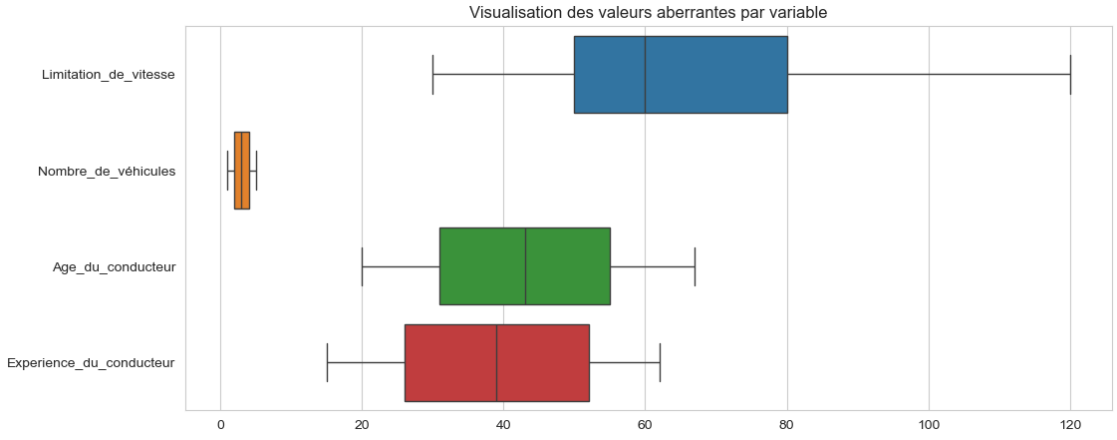
[33]:
```
plt.figure(figsize=(12, 5))
sns.boxplot(data=df.select_dtypes(include='number'), orient='h')
plt.title("Visualisation des valeurs aberrantes par variable")
plt.show()
```



– **Après traitement**

[34]:
```
for col in df.select_dtypes(include='number'):
    df[col] = winsorize(df[col], limits=[0.05, 0.05])
```

[35]:
```
plt.figure(figsize=(12, 5))
sns.boxplot(data=df.select_dtypes(include='number'), orient='h')
plt.title("Visualisation des valeurs aberrantes par variable")
plt.show()
```



Conclusion partielle Traitement de donnée La base de données contient 840 enregistrements avec 14 variables, représentant des éléments qualitatifs (type de route, météo, heure du jour, alcool au volant, gravité, etc.) et quantitatifs (âge et expérience du conducteur, nombre de véhicules impliqués, vitesse limite, etc.). Un taux global de valeurs manquantes de 5% a été identifié et traité avec des méthodes d'imputation adaptées (médiane pour quantitatifs, mode pour qualitatifs), assurant la fiabilité des données analysées. Le nettoyage a permis aussi de gérer les valeurs

aberrantes, garantissant une base prête pour l'analyse statistique.

3 II- Statistiques descriptives

3.1 - Variables quantitatives

3.1.1 - Resumé statistique

[36]:
```
df.describe().transpose()
```

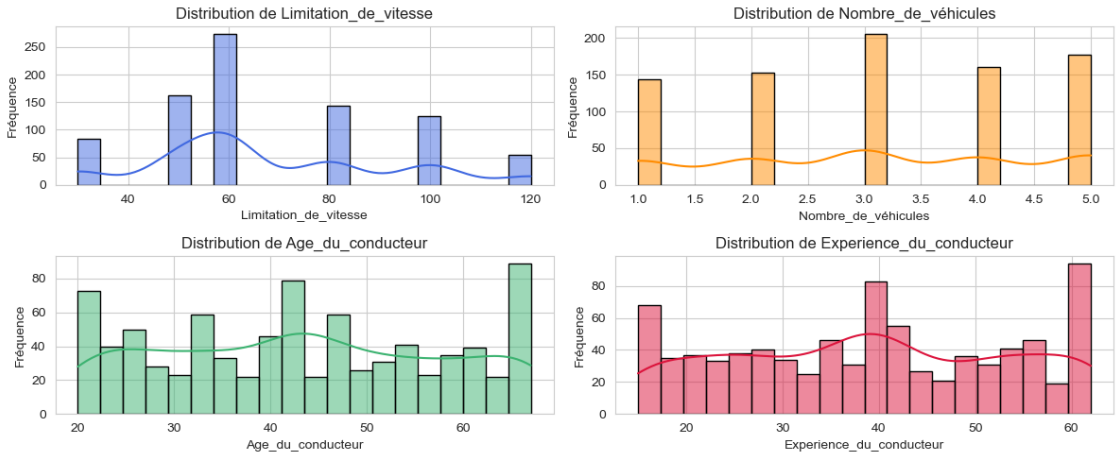
[36]:

	count	mean	std	min	25%	50%	75%	max
Limitation_de_vitesse	840.0	68.238095	24.037921	30.0	50.0	60.0	80.0	120.0
Nombre_de_véhicules	840.0	3.088095	1.377272	1.0	2.0	3.0	4.0	5.0
Age_du_conducteur	840.0	43.228571	14.534170	20.0	31.0	43.0	55.0	67.0
Experience_du_conducteur	840.0	38.953571	14.463352	15.0	26.0	39.0	52.0	62.0

- La vitesse moyenne est limitée à 68 km/h
- Le nombre de véhicule moyen impliqué dans un accident est de 3 véhicules
- L'âge moyen d'un conducteur impliqué dans un accidnt est 43 ans
- En moyenne, le conducteur a une experience de conduite de 38 ans

3.1.2 - Visualisation

[37]:
```
variables = {
    "Limitation_de_vitesse": "royalblue",
    "Nombre_de_véhicules": "darkorange",
    "Age_du_conducteur": "mediumseagreen",
    "Experience_du_conducteur": "crimson"
}

fig, axes = plt.subplots(2, 2, figsize=(12, 5))
axes = axes.flatten()

for i, (var, color) in enumerate(variables.items()):
    sns.histplot(df[var], kde=True, bins=20, ax=axes[i], color=color,␣
 ↪edgecolor='black')
    axes[i].set_title(f'Distribution de {var}', fontsize=12)
    axes[i].set_xlabel(var)
    axes[i].set_ylabel('Fréquence')

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



3.2 - Variables qualitatives

3.2.1 - Resumé statistique

[38]:
```
df.select_dtypes(include = 'object').describe().transpose()
```

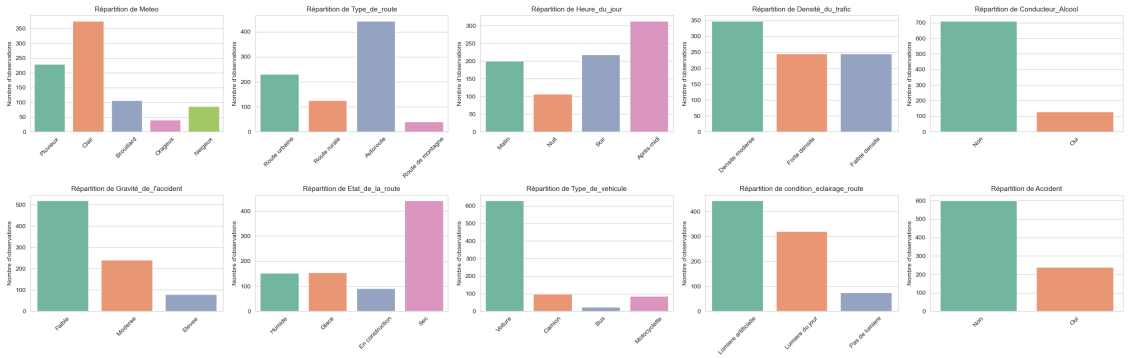
[38]:

	count	unique	top	freq
Meteo	840	5	Clair	376
Type_de_route	840	4	Autoroute	444
Heure_du_jour	840	4	Après-midi	314
Densité_du_trafic	840	3	Densite moderee	349
Conducteur_Alcool	840	2	Non	712
Gravité_de_l'accident	840	3	Faible	520
Etat_de_la_route	840	4	Sec	442
Type_de_vehicule	840	4	Voiture	631
condition_eclairage_route	840	3	Lumiere artificielle	444
Accident	840	2	Non	601

3.2.2 - Visualistation

[39]:
```
qual_vars = [
    "Meteo", "Type_de_route", "Heure_du_jour", "Densité_du_trafic",
    "Conducteur_Alcool", "Gravité_de_l'accident", "Etat_de_la_route",
    "Type_de_vehicule", "condition_eclairage_route", "Accident"
]

n_rows, n_cols = 2, 5
fig, axes = plt.subplots(n_rows, n_cols, figsize=(5 * n_cols, 4 * n_rows))
axes = axes.flatten()

sns.set_style("whitegrid")
```

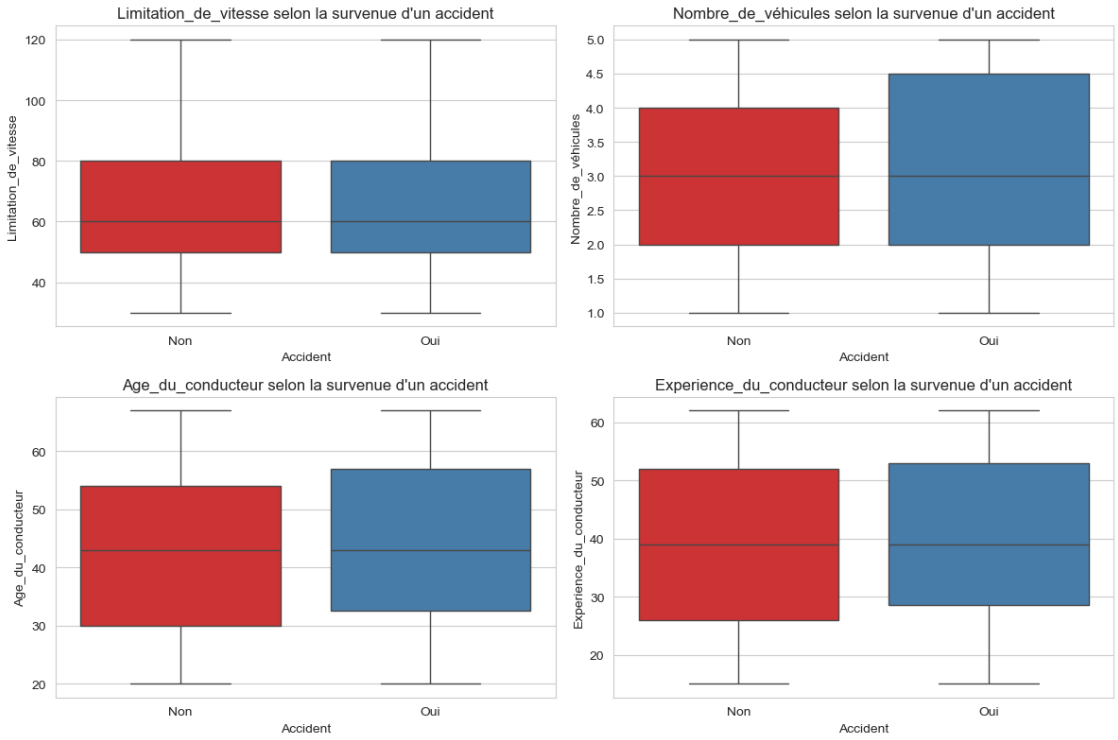
```
palette = sns.color_palette("Set2")

for i, var in enumerate(qual_vars):
    sns.countplot(data=df, x=var, ax=axes[i], palette=palette)
    axes[i].set_title(f"Répartition de {var}", fontsize=11)
    axes[i].set_xlabel("")
    axes[i].set_ylabel("Nombre d'observations")
    axes[i].tick_params(axis='x', rotation=45)

for j in range(len(qual_vars), len(axes)):
    fig.delaxes(axes[j])

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



3.3 - Analyse croisée variable Accident et Variables quantitatives

3.3.1 - Visualisation

[40]:
```
quant_vars = [
    "Limitation_de_vitesse", "Nombre_de_véhicules",
    "Age_du_conducteur", "Experience_du_conducteur"
]

plt.figure(figsize=(12, 8))
for i, var in enumerate(quant_vars, 1):
    plt.subplot(2, 2, i)
    sns.boxplot(data=df, x="Accident", y=var, palette="Set1")
    plt.title(f"{var} selon la survenue d'un accident")
    plt.xlabel("Accident")
    plt.ylabel(var)

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



À travers ces boxplots, on peut supposer qu'il n'existe pas de liaison entre les différentes variables quantitatives et la variable qualitative Accident. Pour en avoir la confirmation, il convient de recourir à un test statistique permettant d'évaluer la relation entre variables quantitatives et qualitatives.

3.3.2 - Tableau

[41]:
```
df.groupby("Accident")[quant_vars].mean().round(2)
```

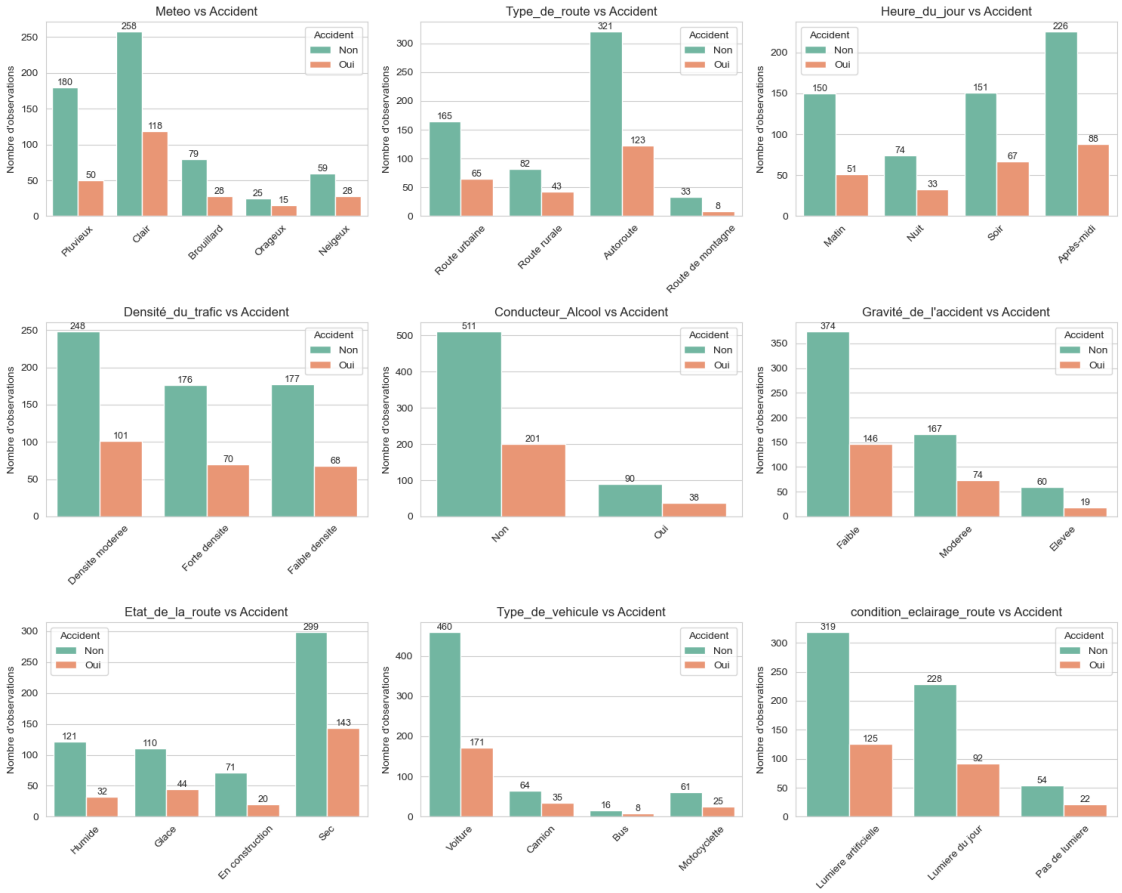
[41]:
```
          Limitation_de_vitesse  Nombre_de_véhicules  Age_du_conducteur  \
Accident
Non                       68.80                 3.05              42.93
Oui                       66.82                 3.18              43.98

          Experience_du_conducteur
Accident
Non                          38.66
Oui                          39.69
```

Parmis les personnes accidentées : - la limitation de vitesse moyenne etait de 67 Km/h - le nombre moyen de véhicules impliqués est de 3 - l'age moyen des conducteur est de 44 ans - l'Experience du conducteur est de 40 ans

3.4 - Analyse croisée variable Accident et Variables qualitatives

3.4.1 - Visualisation

[42]:
```
qual_vars = [
    "Meteo", "Type_de_route", "Heure_du_jour", "Densité_du_trafic",
    "Conducteur_Alcool", "Gravité_de_l'accident", "Etat_de_la_route",
    "Type_de_vehicule", "condition_eclairage_route"
]

n_cols = 3
n_rows = math.ceil(len(qual_vars) / n_cols)

plt.figure(figsize=(5 * n_cols, 4 * n_rows))
sns.set_style("whitegrid")
palette = sns.color_palette("Set2")

for i, var in enumerate(qual_vars, 1):
    ax = plt.subplot(n_rows, n_cols, i)
    plot = sns.countplot(data=df, x=var, hue="Accident", palette=palette, ax=ax)
    ax.set_title(f"{var} vs Accident")
    ax.set_xlabel("")
    ax.set_ylabel("Nombre d'observations")
    ax.tick_params(axis='x', rotation=45)

    for container in ax.containers:
        ax.bar_label(container, fmt='%d', label_type='edge', fontsize=9)

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



3.4.2 - Tableau

[43]:
```
for var in qual_vars:
    print(f"\n--- {var} ---")
    ct = pd.crosstab(df[var], df["Accident"], normalize='index').round(2)
    print(ct)
```

```
--- Meteo ---
Accident      Non   Oui
Meteo
Brouillard   0.74  0.26
Clair        0.69  0.31
Neigeux      0.68  0.32
Orageux      0.62  0.38
Pluvieux     0.78  0.22

--- Type_de_route ---
Accident             Non   Oui
```

```
Type_de_route
Autoroute          0.72  0.28
Route de montagne  0.80  0.20
Route rurale       0.66  0.34
Route urbaine      0.72  0.28

--- Heure_du_jour ---
Accident        Non   Oui
Heure_du_jour
Après-midi     0.72  0.28
Matin          0.75  0.25
Nuit           0.69  0.31
Soir           0.69  0.31

--- Densité_du_trafic ---
Accident            Non   Oui
Densité_du_trafic
Densite moderee    0.71  0.29
Faible densite     0.72  0.28
Forte densite      0.72  0.28

--- Conducteur_Alcool ---
Accident            Non   Oui
Conducteur_Alcool
Non                0.72  0.28
Oui                0.70  0.30

--- Gravité_de_l'accident ---
Accident                Non   Oui
Gravité_de_l'accident
Elevee                 0.76  0.24
Faible                 0.72  0.28
Moderee                0.69  0.31

--- Etat_de_la_route ---
Accident           Non   Oui
Etat_de_la_route
En construction   0.78  0.22
Glace             0.71  0.29
Humide            0.79  0.21
Sec               0.68  0.32

--- Type_de_vehicule ---
Accident           Non   Oui
Type_de_vehicule
Bus               0.67  0.33
Camion            0.65  0.35
Motocyclette      0.71  0.29
```

```
Voiture           0.73  0.27

--- condition_eclairage_route ---
Accident                    Non   Oui
condition_eclairage_route
Lumiere artificielle       0.72  0.28
Lumiere du jour            0.71  0.29
Pas de lumiere             0.71  0.29
```

Parmi les personnes ayant eu un accident : - 38 % étaient exposées à des conditions météorologiques orageuses. - 34 % circulaient sur une route rurale. - 31 % ont eu leur accident en soirée ou durant la nuit. - 29 % évoluaient dans un trafic de densité modérée. - 30 % avaient consommé de l'alcool. - 31 % ont subi un accident de gravité modérée. - 32 % ont eu un accident sur une chaussée sèche. - 35 % étaient impliquées dans un accident avec un camion. - 29 % ont eu un accident dans des conditions d'éclairage comprenant soit la lumière du jour, soit l'absence totale de lumière

Conclusion partielle analyse descriptive L'analyse descriptive confirme des caractéristiques cohérentes : l'âge moyen du conducteur est de 43 ans, l'expérience moyenne est de 39 ans, et le nombre moyen de véhicules impliqués dans un accident est de 3. Sur le plan qualitatif, la majorité des accidents surviennent sous conditions météorologiques claires (45%), sur autoroute (52%), avec une densité de trafic modérée (41%). Ces premières observations posent les bases pour la recherche d'associations plus spécifiques avec la variable d'intérêt, la survenue d'accident

4 III- Tests Statistiques

4.1 –Variable Accident et Variables qualitatives

[44]:
```
qual_vars = [
    "Meteo", "Type_de_route", "Heure_du_jour", "Densité_du_trafic",
    "Conducteur_Alcool", "Gravité_de_l'accident", "Etat_de_la_route",
    "Type_de_vehicule", "condition_eclairage_route"
]

for var in qual_vars:
    print(f"\n  Test entre Accident et {var}")

    table = pd.crosstab(df["Accident"], df[var])
    print("  Tableau observé :")
    print(table)

    chi2, p_chi2, dof, expected = chi2_contingency(table)
    expected_df = pd.DataFrame(expected, index=table.index, columns=table.
 ↪columns)

    print("\n  Effectifs théoriques :")
    print(expected_df.round(2))
```

```
    cochran_mask = expected >= 5
    cochran_pct = round(cochran_mask.sum().sum() / expected.size * 100, 2)
    cochran_ok = cochran_pct == 100
    print(f"\n Cochran : {cochran_pct}% des cellules  5 → {'OK' if cochran_ok␣
↪else '  NON RESPECTÉ'}")

    if cochran_ok:
        print(f" Test du Khi-deux appliqué : p = {p_chi2:.4f}")
        if p_chi2 < 0.05:
            print(f" Conclusion : Liaison significative entre Accident et␣
↪{var}")
        else:
            print(f" Conclusion : Aucune liaison significative détectée entre␣
↪Accident et {var}")
    elif table.shape == (2, 2):
        fisher_stat, fisher_p = fisher_exact(table)
        print(f" Test de Fisher exact appliqué : p = {fisher_p:.4f}")
        if fisher_p < 0.05:
            print(f" Conclusion : Liaison significative entre Accident et␣
↪{var}")
        else:
            print(f" Conclusion : Aucune liaison significative détectée entre␣
↪Accident et {var}")
    else:
        print(" Tableau trop complexe pour Fisher et Cochran non respectée →␣
↪test non fiable")
```

```
 Test entre Accident et Meteo
 Tableau observé :
Meteo     Brouillard  Clair  Neigeux  Orageux  Pluvieux
Accident
Non               79    258       59       25       180
Oui               28    118       28       15        50

 Effectifs théoriques :
Meteo     Brouillard   Clair  Neigeux  Orageux  Pluvieux
Accident
Non            76.56  269.02    62.25    28.62    164.56
Oui            30.44  106.98    24.75    11.38     65.44

 Cochran : 100.0% des cellules  5 → OK
 Test du Khi-deux appliqué : p = 0.0573
 Conclusion : Aucune liaison significative détectée entre Accident et Meteo

 Test entre Accident et Type_de_route
 Tableau observé :
```

```
Type_de_route  Autoroute  Route de montagne  Route rurale  Route urbaine
Accident
Non                321                 33            82            165
Oui                123                  8            43             65

 Effectifs théoriques :
Type_de_route  Autoroute  Route de montagne  Route rurale  Route urbaine
Accident
Non               317.67              29.33         89.43         164.56
Oui               126.33              11.67         35.57          65.44

 Cochran : 100.0% des cellules  5 → OK
 Test du Khi-deux appliqué : p = 0.2715
 Conclusion : Aucune liaison significative détectée entre Accident et
Type_de_route

 Test entre Accident et Heure_du_jour
 Tableau observé :
Heure_du_jour  Après-midi  Matin  Nuit  Soir
Accident
Non                  226    150    74   151
Oui                   88     51    33    67

 Effectifs théoriques :
Heure_du_jour  Après-midi   Matin   Nuit    Soir
Accident
Non                224.66  143.81  76.56  155.97
Oui                 89.34   57.19  30.44   62.03

 Cochran : 100.0% des cellules  5 → OK
 Test du Khi-deux appliqué : p = 0.6102
 Conclusion : Aucune liaison significative détectée entre Accident et
Heure_du_jour

 Test entre Accident et Densité_du_trafic
 Tableau observé :
Densité_du_trafic  Densite moderee  Faible densite  Forte densite
Accident
Non                           248             177            176
Oui                           101              68             70

 Effectifs théoriques :
Densité_du_trafic  Densite moderee  Faible densite  Forte densite
Accident
Non                          249.7          175.29         176.01
Oui                           99.3           69.71          69.99

 Cochran : 100.0% des cellules  5 → OK
```

```
  Test du Khi-deux appliqué : p = 0.9516
  Conclusion : Aucune liaison significative détectée entre Accident et
Densité_du_trafic

  Test entre Accident et Conducteur_Alcool
  Tableau observé :
Conducteur_Alcool  Non  Oui
Accident
Non                511   90
Oui                201   38

  Effectifs théoriques :
Conducteur_Alcool     Non    Oui
Accident
Non                509.42  91.58
Oui                202.58  36.42

  Cochran : 100.0% des cellules  5 → OK
  Test du Khi-deux appliqué : p = 0.8181
  Conclusion : Aucune liaison significative détectée entre Accident et
Conducteur_Alcool

  Test entre Accident et Gravité_de_l'accident
  Tableau observé :
Gravité_de_l'accident  Elevee  Faible  Moderee
Accident
Non                       60     374      167
Oui                       19     146       74

  Effectifs théoriques :
Gravité_de_l'accident  Elevee  Faible  Moderee
Accident
Non                    56.52  372.05   172.43
Oui                    22.48  147.95    68.57

  Cochran : 100.0% des cellules  5 → OK
  Test du Khi-deux appliqué : p = 0.4994
  Conclusion : Aucune liaison significative détectée entre Accident et
Gravité_de_l'accident

  Test entre Accident et Etat_de_la_route
  Tableau observé :
Etat_de_la_route  En construction  Glace  Humide  Sec
Accident
Non                          71    110     121  299
Oui                          20     44      32  143

  Effectifs théoriques :
```

```
Etat_de_la_route  En construction   Glace  Humide     Sec
Accident
Non                         65.11  110.18  109.47  316.24
Oui                         25.89   43.82   43.53  125.76

  Cochran : 100.0% des cellules  5 → OK
  Test du Khi-deux appliqué : p = 0.0239
  Conclusion : Liaison significative entre Accident et Etat_de_la_route

  Test entre Accident et Type_de_vehicule
  Tableau observé :
Type_de_vehicule  Bus  Camion  Motocyclette  Voiture
Accident
Non                16      64            61      460
Oui                 8      35            25      171

  Effectifs théoriques :
Type_de_vehicule    Bus  Camion  Motocyclette  Voiture
Accident
Non               17.17   70.83         61.53   451.47
Oui                6.83   28.17         24.47   179.53

  Cochran : 100.0% des cellules  5 → OK
  Test du Khi-deux appliqué : p = 0.3647
  Conclusion : Aucune liaison significative détectée entre Accident et
Type_de_vehicule

  Test entre Accident et condition_eclairage_route
  Tableau observé :
condition_eclairage_route  Lumiere artificielle  Lumiere du jour  \
Accident
Non                                          319              228
Oui                                          125               92

condition_eclairage_route  Pas de lumiere
Accident
Non                                    54
Oui                                    22

  Effectifs théoriques :
condition_eclairage_route  Lumiere artificielle  Lumiere du jour  \
Accident
Non                                     317.67           228.95
Oui                                     126.33            91.05

condition_eclairage_route  Pas de lumiere
Accident
Non                                54.38
```

```
Oui                                    21.62

 Cochran : 100.0% des cellules  5 → OK
 Test du Khi-deux appliqué : p = 0.9789
 Conclusion : Aucune liaison significative détectée entre Accident et
condition_eclairage_route
```

4.2 –Variable Accident et Variables quantitatives

[45]:
```python
quant_vars = [
    "Limitation_de_vitesse", "Nombre_de_véhicules",
    "Age_du_conducteur", "Experience_du_conducteur"
]

for var in quant_vars:
    print(f"\n Test entre {var} et Accident")
    groups = [df[df["Accident"] == cat][var].dropna() for cat in df["Accident"].
 ↪unique()]

    normal = all(shapiro(g)[1] > 0.05 for g in groups)

    if normal:
        var_test_p = bartlett(*groups)[1]
        var_test_name = "Bartlett"
    else:
        var_test_p = levene(*groups)[1]
        var_test_name = "Levene"

    equal_var = var_test_p > 0.05

    if normal and equal_var:
        stat, p = f_oneway(*groups)
        test_name = "ANOVA"
    else:
        stat, p = kruskal(*groups)
        test_name = "Kruskal-Wallis"

    print(f" Normalité : {normal}")
    print(f" Test de variances : {var_test_name} | p = {var_test_p:.4f} →␣
 ↪Variances égales : {equal_var}")
    print(f" Test utilisé : {test_name} | p = {p:.4f}")

    if p < 0.05:
        print(f" Conclusion : Liaison significative entre {var} et Accident")
    else:
        print(f" Conclusion : Aucune liaison significative détectée entre␣
 ↪{var} et Accident")
```

```
 Test entre Limitation_de_vitesse et Accident
 Normalité : False
 Test de variances : Levene | p = 0.3143 → Variances égales : True
 Test utilisé : Kruskal-Wallis | p = 0.3324
 Conclusion : Aucune liaison significative détectée entre Limitation_de_vitesse
et Accident

 Test entre Nombre_de_véhicules et Accident
 Normalité : False
 Test de variances : Levene | p = 0.0775 → Variances égales : True
 Test utilisé : Kruskal-Wallis | p = 0.2344
 Conclusion : Aucune liaison significative détectée entre Nombre_de_véhicules
et Accident

 Test entre Age_du_conducteur et Accident
 Normalité : False
 Test de variances : Levene | p = 0.4493 → Variances égales : True
 Test utilisé : Kruskal-Wallis | p = 0.3762
 Conclusion : Aucune liaison significative détectée entre Age_du_conducteur et
Accident

 Test entre Experience_du_conducteur et Accident
 Normalité : False
 Test de variances : Levene | p = 0.3004 → Variances égales : True
 Test utilisé : Kruskal-Wallis | p = 0.4052
 Conclusion : Aucune liaison significative détectée entre
Experience_du_conducteur et Accident
```

Conclusion partielle Tests statistiques Les tests d'association ont révélé une seule liaison statistiquement significative entre la survenue d'accident et l'état de la route (p=0.024), particulièrement les routes sèches qui augmentent les risques comparé aux routes en construction. Aucune association significative n'a été détectée avec les autres variables qualitatives (météo, type de route, alcool, gravité, éclairage) ni avec les variables quantitatives (âge, expérience, vitesse limite, nombre de véhicules). Ces résultats suggèrent que l'état de la route est un facteur clé influençant les accidents, nécessitant une modélisation plus fine.

5 IV- Regression logistique

5.1 - Modélisation

Il s'agit ici de modéliser le risque d'accident (Variables qualitatives binaires) en fonction des autres variables

[46]:
```
X = pd.get_dummies(df.drop(columns=["Accident"]), drop_first=True)

X = X.astype(float)
```

```python
X = sm.add_constant(X)

y = df["Accident"].map({"Non": 0, "Oui": 1})
y = y.astype(float)

model = sm.Logit(y, X).fit()

print(model.summary())
```

```
Optimization terminated successfully.
         Current function value: 0.577348
         Iterations 5
                           Logit Regression Results
==============================================================================
Dep. Variable:               Accident   No. Observations:                  840
Model:                          Logit   Df Residuals:                      812
Method:                           MLE   Df Model:                           27
Date:                Mon, 29 Sep 2025   Pseudo R-squ.:                0.03320
Time:                        22:33:20   Log-Likelihood:                -484.97
converged:                       True   LL-Null:                       -501.63
Covariance Type:            nonrobust   LLR p-value:                    0.1870
==============================================================================
                                                 coef    std err          z
P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
------------------------------
const                                       -1.6996      0.738     -2.303
0.021      -3.146      -0.253
Limitation_de_vitesse                       -0.0040      0.003     -1.206
0.228      -0.011       0.003
Nombre_de_véhicules                          0.0683      0.058      1.178
0.239      -0.045       0.182
Age_du_conducteur                           -0.0010      0.015     -0.070
0.944      -0.030       0.028
Experience_du_conducteur                     0.0072      0.015      0.484
0.629      -0.022       0.036
Meteo_Clair                                  0.2133      0.251      0.848
0.396      -0.280       0.706
Meteo_Neigeux                                0.2664      0.326      0.818
0.413      -0.372       0.905
Meteo_Orageux                                0.5555      0.401      1.384
0.166      -0.231       1.342
Meteo_Pluvieux                              -0.2706      0.277     -0.978
0.328      -0.813       0.272
Type_de_route_Route de montagne             -0.5274      0.419     -1.257
0.209      -1.349       0.295
```

```
Type_de_route_Route rurale                    0.2955      0.221      1.336
0.182      -0.138       0.729
Type_de_route_Route urbaine                   0.0625      0.186      0.336
0.737      -0.302       0.427
Heure_du_jour_Matin                          -0.0633      0.212     -0.298
0.766      -0.480       0.353
Heure_du_jour_Nuit                            0.1172      0.251      0.467
0.640      -0.374       0.609
Heure_du_jour_Soir                            0.1392      0.200      0.698
0.485      -0.252       0.530
Densité_du_trafic_Faible densite             -0.0492      0.191     -0.257
0.797      -0.424       0.325
Densité_du_trafic_Forte densite              -0.0332      0.191     -0.174
0.862      -0.408       0.341
Conducteur_Alcool_Oui                         0.0253      0.218      0.116
0.907      -0.401       0.452
Gravité_de_l'accident_Faible                  0.2505      0.292      0.857
0.392      -0.322       0.823
Gravité_de_l'accident_Moderee                 0.3568      0.310      1.152
0.249      -0.250       0.964
Etat_de_la_route_Glace                        0.4012      0.318      1.262
0.207      -0.222       1.025
Etat_de_la_route_Humide                      -0.0660      0.331     -0.199
0.842      -0.715       0.583
Etat_de_la_route_Sec                          0.5778      0.281      2.057
0.040       0.027       1.128
Type_de_vehicule_Camion                       0.1670      0.499      0.334
0.738      -0.812       1.146
Type_de_vehicule_Motocyclette                -0.1065      0.512     -0.208
0.835      -1.111       0.898
Type_de_vehicule_Voiture                     -0.2315      0.459     -0.504
0.614      -1.132       0.669
condition_eclairage_route_Lumiere du jour    -0.0394      0.168     -0.234
0.815      -0.370       0.291
condition_eclairage_route_Pas de lumiere     -0.0680      0.284     -0.239
0.811      -0.625       0.489
==============================================================================
==============================
```

La constante et la modalité Sec de la variable Etat_de_la route sont les seuls significatifs. Vérifions si les variables elle meme sont significatives avec l'anova

5.2 - Faire l'anova du modèle

[47]:
```
from scipy.stats import chi2
import statsmodels.api as sm

# Liste des variables dans l'ordre d'ajout
```

```python
vars_seq = [
    "Meteo", "Type_de_route", "Heure_du_jour", "Densité_du_trafic",
    "Limitation_de_vitesse", "Nombre_de_véhicules", "Conducteur_Alcool",
    "Gravité_de_l'accident", "Etat_de_la_route", "Type_de_vehicule",
    "Age_du_conducteur", "Experience_du_conducteur", "condition_eclairage_route"
]

# Variable cible
y = df["Accident"].map({"Non": 0, "Oui": 1}).astype(float)

# Modèle nul (intercept seul)
X_null = sm.add_constant(pd.DataFrame(index=df.index))
model_null = sm.Logit(y, X_null).fit(disp=0)
ll_prev = model_null.llf
df_prev = model_null.df_model
resid_dev_prev = -2 * ll_prev

print(f"{'Variable':<30} {'Df':>3} {'Deviance':>10} {'Resid. Df':>10} {'Resid.␣
 ↪Dev':>12} {'Pr(>Chi)':>10}")
print("-" * 80)

# Construction séquentielle
X_seq = pd.DataFrame(index=df.index)

for var in vars_seq:
    # Ajout de la variable encodée
    new_cols = pd.get_dummies(df[var], drop_first=True)
    X_seq = pd.concat([X_seq, new_cols], axis=1)
    X_model = sm.add_constant(X_seq.astype(float))

    # Modèle avec la variable ajoutée
    model = sm.Logit(y, X_model).fit(disp=0)
    ll_new = model.llf
    df_new = model.df_model
    resid_dev_new = -2 * ll_new

    # Calculs
    df_diff = df_new - df_prev
    deviance = resid_dev_prev - resid_dev_new
    p_value = chi2.sf(deviance, df_diff)

    # Affichage
    print(f"{var:<30} {df_diff:>3} {deviance:>10.4f} {df_new:>10}␣
 ↪{resid_dev_new:>12.2f} {p_value:>10.5f}")

    # Mise à jour
    ll_prev = ll_new
```

```
    df_prev = df_new
    resid_dev_prev = resid_dev_new
```

```
Variable                         Df    Deviance  Resid. Df    Resid. Dev   Pr(>Chi)
-------------------------------------------------------------------------------
Meteo                           4.0      9.3131        4.0        993.94    0.05373
Type_de_route                   3.0      3.4646        7.0        990.48    0.32538
Heure_du_jour                   3.0      1.9261       10.0        988.55    0.58788
Densité_du_trafic               2.0      0.0451       12.0        988.51    0.97770
Limitation_de_vitesse           5.0      1.6843       17.0        986.82    0.89087
Nombre_de_véhicules             4.0      3.8806       21.0        982.94    0.42241
Conducteur_Alcool               1.0      0.0301       22.0        982.91    0.86217
Gravité_de_l'accident           2.0      1.2102       24.0        981.70    0.54601
Etat_de_la_route                3.0     10.7808       27.0        970.92    0.01297
Type_de_vehicule                3.0      2.7793       30.0        968.14    0.42692
Age_du_conducteur              47.0     51.8700       77.0        916.27
0.28972
Experience_du_conducteur       47.0     36.2892      124.0        879.98
0.87112
condition_eclairage_route       2.0      0.0531      126.0        879.93    0.97382
```

les variables Meteo et Etat_de_la_route sont significatives et donc serviront d'interpretation.

5.3 - Calcul des odd-ratio

[48]:
```
X = sm.add_constant(X)
model_full = sm.Logit(y, X).fit()

coefficients = model_full.params
odds_ratios = np.exp(coefficients)

odds_ratios
```

```
Optimization terminated successfully.
         Current function value: 0.577348
         Iterations 5
```

[48]:
```
const                                  0.182761
Limitation_de_vitesse                  0.995964
Nombre_de_véhicules                    1.070643
Age_du_conducteur                      0.998970
Experience_du_conducteur               1.007207
Meteo_Clair                            1.237699
Meteo_Neigeux                          1.305285
Meteo_Orageux                          1.742808
Meteo_Pluvieux                         0.762918
Type_de_route_Route de montagne        0.590151
Type_de_route_Route rurale             1.343823
```

```
Type_de_route_Route urbaine                    1.064465
Heure_du_jour_Matin                            0.938708
Heure_du_jour_Nuit                             1.124296
Heure_du_jour_Soir                             1.149399
Densité_du_trafic_Faible densite               0.952008
Densité_du_trafic_Forte densite                0.967324
Conducteur_Alcool_Oui                          1.025653
Gravité_de_l'accident_Faible                   1.284620
Gravité_de_l'accident_Moderee                  1.428687
Etat_de_la_route_Glace                         1.493627
Etat_de_la_route_Humide                        0.936152
Etat_de_la_route_Sec                           1.782049
Type_de_vehicule_Camion                        1.181722
Type_de_vehicule_Motocyclette                  0.898945
Type_de_vehicule_Voiture                       0.793348
condition_eclairage_route_Lumiere du jour      0.961368
condition_eclairage_route_Pas de lumiere       0.934289
dtype: float64
```

OR(Accident/Etat_de_la_route_Sec) = 1.782049 Par rapport à un etat de la route en construction, une etat de la route sec multiplie par 2 les chances d'avoir un accident.

OR(Accident/const) = 0.182761 Toutes choses étant égales par ailleurs, la probabilité qu'un accident survienne est multipliée par 1.20 par rapport à la situation de référence.

5.4 - Calcul des effets marginaux

ils mesurent la variation de la probabilité de l'événement associé à une augmentation marginale d'une unité de la variable indépendante, tout en maintenant des constantes les autres variables du modèle

[49]:
```
model = sm.Logit(y, X).fit()

mfx = model.get_margeff()
print(mfx.summary())
```

```
Optimization terminated successfully.
         Current function value: 0.577348
         Iterations 5
        Logit Marginal Effects
=====================================
Dep. Variable:               Accident
Method:                          dydx
At:                           overall
================================================================================
==============================
                                                dy/dx    std err          z
P>|z|      [0.025      0.975]
--------------------------------------------------------------------------------
```

```
-----------------------------
Limitation_de_vitesse                    -0.0008      0.001     -1.209
0.227      -0.002       0.000
Nombre_de_véhicules                       0.0134      0.011      1.181
0.238      -0.009       0.036
Age_du_conducteur                        -0.0002      0.003     -0.070
0.944      -0.006       0.005
Experience_du_conducteur                  0.0014      0.003      0.484
0.628      -0.004       0.007
Meteo_Clair                               0.0417      0.049      0.849
0.396      -0.055       0.138
Meteo_Neigeux                             0.0521      0.064      0.819
0.413      -0.073       0.177
Meteo_Orageux                             0.1087      0.078      1.390
0.165      -0.045       0.262
Meteo_Pluvieux                           -0.0529      0.054     -0.980
0.327      -0.159       0.053
Type_de_route_Route de montagne          -0.1032      0.082     -1.261
0.207      -0.264       0.057
Type_de_route_Route rurale                0.0578      0.043      1.341
0.180      -0.027       0.142
Type_de_route_Route urbaine               0.0122      0.036      0.336
0.737      -0.059       0.083
Heure_du_jour_Matin                      -0.0124      0.042     -0.298
0.766      -0.094       0.069
Heure_du_jour_Nuit                        0.0229      0.049      0.467
0.640      -0.073       0.119
Heure_du_jour_Soir                        0.0272      0.039      0.698
0.485      -0.049       0.104
Densité_du_trafic_Faible densite         -0.0096      0.037     -0.257
0.797      -0.083       0.064
Densité_du_trafic_Forte densite          -0.0065      0.037     -0.174
0.862      -0.080       0.067
Conducteur_Alcool_Oui                     0.0050      0.043      0.116
0.907      -0.078       0.088
Gravité_de_l'accident_Faible              0.0490      0.057      0.858
0.391      -0.063       0.161
Gravité_de_l'accident_Moderee             0.0698      0.060      1.155
0.248      -0.049       0.188
Etat_de_la_route_Glace                    0.0785      0.062      1.265
0.206      -0.043       0.200
Etat_de_la_route_Humide                  -0.0129      0.065     -0.199
0.842      -0.140       0.114
Etat_de_la_route_Sec                      0.1130      0.055      2.073
0.038       0.006       0.220
Type_de_vehicule_Camion                   0.0327      0.098      0.334
0.738      -0.159       0.224
Type_de_vehicule_Motocyclette            -0.0208      0.100     -0.208
```

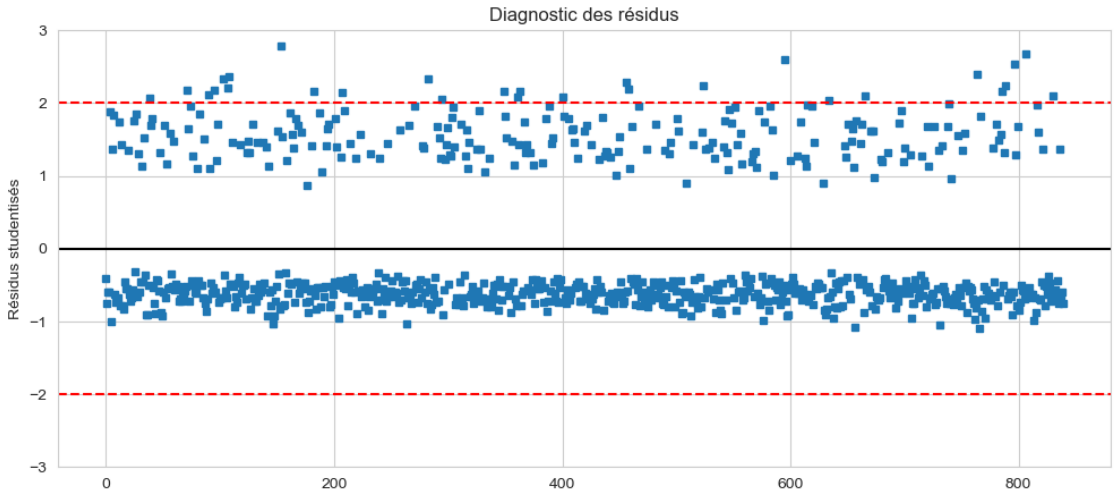
```
0.835      -0.217       0.176
Type_de_vehicule_Voiture                        -0.0453      0.090     -0.504
0.614      -0.221       0.131
condition_eclairage_route_Lumiere du jour       -0.0077      0.033     -0.234
0.815      -0.072       0.057
condition_eclairage_route_Pas de lumiere        -0.0133      0.056     -0.239
0.811      -0.122       0.096
==============================================================================
==============================
```

Par rapport à un etat de la route en construction, la probabilité de risque d'accident augmente d'environ 0,1130

5.5 - Analyse des résidus

En théorie 95% des résidus studentisés se trouvent dans l'intervalle [-2;2].

[50]:
```
influence = model_full.get_influence()
res_studentized = influence.resid_studentized

plt.figure(figsize=(12, 5))
plt.plot(res_studentized, marker='s', linestyle='none', markersize=4)
plt.axhline(y=0, linestyle='-', color='black')
plt.axhline(y=2, linestyle='--', color='red')
plt.axhline(y=-2, linestyle='--', color='red')
plt.ylim(-3, 3)
plt.ylabel("Résidus studentisés")
plt.title("Diagnostic des résidus")
plt.show()
```



[51]:
```
influence = model_full.get_influence()
res_studentized = influence.resid_studentized

pourcentage = (np.abs(res_studentized) <= 3).mean() * 100

print(f"  Pourcentage d'observations bien modélisées : {pourcentage:.2f}%")
```

```
 Pourcentage d'observations bien modélisées : 100.00%
```

5.6 - Taux de mauvais classement et Matrice de Confusion

- **Calcul des probabilité**

[52]:
```
df["PROBABILITE_PREDITE"] = model_full.predict()
df.head()
```

[52]:

	Meteo	Type_de_route	Heure_du_jour	Densité_du_trafic	Limitation_de_vitesse	Nombre_de_véhicules	Conducteur_Alcool	Gravité_de_l'accident	Etat_de_la_route	Type_de_vehicule	Age_du_conducteur	Experience_du_conducteur	condition_eclairage_route	Accident	PROBABILITE_PREDITE
0	Pluvieux	Route urbaine	Matin	Densite moderee	100.0	5.0	Non	Faible	Humide	Voiture	51.0	48.0	Lumiere artificielle	Non	0.143197
1	Clair	Route rurale	Nuit	Densite moderee	120.0	3.0	Non	Moderee	Humide	Camion	49.0	43.0	Lumiere artificielle	Non	0.345662
2	Pluvieux	Autoroute	Soir	Densite moderee	60.0	4.0	Non	Faible	Glace	Voiture	54.0	52.0	Lumiere artificielle	Non	0.256804
3	Clair	Route urbaine	Après-midi	Forte densite	60.0	3.0	Non	Faible	En construction	Bus	34.0	31.0	Lumiere du jour	Non	
4	Pluvieux	Autoroute	Matin	Densite moderee	120.0	5.0	Non	Faible	Sec	Voiture	62.0	55.0	Lumiere artificielle	Oui	

```
3              0.250429
4              0.222756
```

- **Transformer les probabilité en modalite predites**

[53]:
```
fpr, tpr, thresholds = roc_curve(df["Accident"].map({"Non": 0, "Oui": 1}),␣
↪df["PROBABILITE_PREDITE"])

youden_index = tpr - fpr
optimal_idx = np.argmax(youden_index)
optimal_threshold = thresholds[optimal_idx]

print(f"  Seuil optimal selon Youden Index : {optimal_threshold:.2f}")
```

```
 Seuil optimal selon Youden Index : 0.32
```

[54]:
```
df["MODALITE_PREDITE"] = np.where(df["PROBABILITE_PREDITE"] < 0.32, "Non",␣
↪"Oui")
df.head()
```

[54]:

	Meteo	Type_de_route	Heure_du_jour	Densité_du_trafic	\
0	Pluvieux	Route urbaine	Matin	Densite moderee	
1	Clair	Route rurale	Nuit	Densite moderee	
2	Pluvieux	Autoroute	Soir	Densite moderee	
3	Clair	Route urbaine	Après-midi	Forte densite	
4	Pluvieux	Autoroute	Matin	Densite moderee	

	Limitation_de_vitesse	Nombre_de_véhicules	Conducteur_Alcool	\
0	100.0	5.0	Non	
1	120.0	3.0	Non	
2	60.0	4.0	Non	
3	60.0	3.0	Non	
4	120.0	5.0	Non	

	Gravité_de_l'accident	Etat_de_la_route	Type_de_vehicule	Age_du_conducteur	\
0	Faible	Humide	Voiture	51.0	
1	Moderee	Humide	Camion	49.0	
2	Faible	Glace	Voiture	54.0	
3	Faible	En construction	Bus	34.0	
4	Faible	Sec	Voiture	62.0	

	Experience_du_conducteur	condition_eclairage_route	Accident	\
0	48.0	Lumiere artificielle	Non	
1	43.0	Lumiere artificielle	Non	
2	52.0	Lumiere artificielle	Non	
3	31.0	Lumiere du jour	Non	
4	55.0	Lumiere artificielle	Oui	

```
   PROBABILITE_PREDITE MODALITE_PREDITE
0             0.143197              Non
1             0.345662              Oui
2             0.256804              Non
3             0.250429              Non
4             0.222756              Non
```

- **Calculer le taux de mauvais classement à partir de la matrice de confusion**

[55]:
```
matrice_confusion = pd.crosstab(df["Accident"], df["MODALITE_PREDITE"])
matrice_confusion
```

[55]:
```
MODALITE_PREDITE  Non  Oui
Accident
Non               425  176
Oui               122  117
```

[56]:
```
VP = 117
VN = 425
FP = 176
FN = 122

tmc_negatif = (FP + FN) / (VP + VN + FP + FN)

print(f"  Taux de mauvais classement : {tmc_negatif:.4f} ({tmc_negatif*100:.
 ↪2f}%)")
```

```
Taux de mauvais classement : 0.3548 (35.48%)
```

- **Prediction pour un nouvel individu en entrant ses caractéristiques**

[57]:
```
nouvel_individu = {
    "Meteo": input("Météo (Ex: Clair, Pluie, Brouillard) : "),
    "Type_de_route": input("Type de route (Ex: Urbain, Rural, Autoroute) : "),
    "Heure_du_jour": input("Heure du jour (Ex: Jour, Nuit) : "),
    "Densité_du_trafic": input("Densité du trafic (Ex: Faible, Moyenne, Élevée)␣
 ↪: "),
    "Limitation_de_vitesse": input("Limitation de vitesse (Ex: 50, 90, 130) :␣
 ↪"),
    "Nombre_de_véhicules": input("Nombre de véhicules impliqués : "),
    "Conducteur_Alcool": input("Conducteur alcoolisé (Oui/Non) : "),
    "Gravité_de_l'accident": input("Gravité (Ex: Léger, Grave, Mortel) : "),
    "Etat_de_la_route": input("État de la route (Ex: Bonne, Dégradée) : "),
    "Type_de_vehicule": input("Type de véhicule (Ex: Voiture, Moto, Camion) :␣
 ↪"),
    "Age_du_conducteur": float(input("Âge du conducteur : ")),
```

```
    "Experience_du_conducteur": float(input("Années d'expérience du conducteur :
 ↪ ")),
    "condition_eclairage_route": input("Éclairage de la route (Ex: Bon, Faible,␣
 ↪Aucun) : ")
}

ind_df = pd.DataFrame([nouvel_individu])
ind_encoded = pd.get_dummies(ind_df)

model_vars = model_full.model.exog_names[1:]

for col in model_vars:
    if col not in ind_encoded.columns:
        ind_encoded[col] = 0

ind_encoded = ind_encoded[model_vars]

ind_encoded = sm.add_constant(ind_encoded, has_constant='add')

ind_encoded = ind_encoded.astype(float)

proba = model_full.predict(ind_encoded)[0]
modalite_predite = "Oui" if proba >= 0.32 else "Non"

print(f"\n Probabilité estimée d'accident : {proba:.4f}")
print(f" Risque d'accident (seuil 0.32) : {modalite_predite}")
```

```
Météo (Ex: Clair, Pluie, Brouillard) :  Pluvieux
Type de route (Ex: Urbain, Rural, Autoroute) :  Route urbaine
Heure du jour (Ex: Jour, Nuit) :  Matin
Densité du trafic (Ex: Faible, Moyenne, Élevée) :  Densite moderee
Limitation de vitesse (Ex: 50, 90, 130) :  100
Nombre de véhicules impliqués :  5
Conducteur alcoolisé (Oui/Non) :  Non
Gravité (Ex: Léger, Grave, Mortel) :  Faible
État de la route (Ex: Bonne, Dégradée) :  Humide
Type de véhicule (Ex: Voiture, Moto, Camion) :  Voiture
Âge du conducteur :  51
Années d'expérience du conducteur :  48
Éclairage de la route (Ex: Bon, Faible, Aucun) :  Lumiere artificielle


 Probabilité estimée d'accident : 0.1511
 Risque d'accident (seuil 0.32) : Non
```

[58]:
```
nouvel_individu = {
    "Meteo": input("Météo (Ex: Clair, Pluie, Brouillard) : "),
```

```
    "Type_de_route": input("Type de route (Ex: Urbain, Rural, Autoroute) : "),
    "Heure_du_jour": input("Heure du jour (Ex: Jour, Nuit) : "),
    "Densité_du_trafic": input("Densité du trafic (Ex: Faible, Moyenne, Élevée)␣
↪: "),
    "Limitation_de_vitesse": input("Limitation de vitesse (Ex: 50, 90, 130) :␣
↪"),
    "Nombre_de_véhicules": input("Nombre de véhicules impliqués : "),
    "Conducteur_Alcool": input("Conducteur alcoolisé (Oui/Non) : "),
    "Gravité_de_l'accident": input("Gravité (Ex: Léger, Grave, Mortel) : "),
    "Etat_de_la_route": input("État de la route (Ex: Bonne, Dégradée) : "),
    "Type_de_vehicule": input("Type de véhicule (Ex: Voiture, Moto, Camion) :␣
↪"),
    "Age_du_conducteur": float(input("Âge du conducteur : ")),
    "Experience_du_conducteur": float(input("Années d'expérience du conducteur :
↪ ")),
    "condition_eclairage_route": input("Éclairage de la route (Ex: Bon, Faible,␣
↪Aucun) : ")
}

ind_df = pd.DataFrame([nouvel_individu])
ind_encoded = pd.get_dummies(ind_df)

model_vars = model_full.model.exog_names[1:]

for col in model_vars:
    if col not in ind_encoded.columns:
        ind_encoded[col] = 0

ind_encoded = ind_encoded[model_vars]

ind_encoded = sm.add_constant(ind_encoded, has_constant='add')

ind_encoded = ind_encoded.astype(float)

proba = model_full.predict(ind_encoded)[0]
modalite_predite = "Oui" if proba >= 0.32 else "Non"

print(f"\n Probabilité estimée d'accident : {proba:.4f}")
print(f" Risque d'accident (seuil 0.32) : {modalite_predite}")
```

```
Météo (Ex: Clair, Pluie, Brouillard) :  Clair
Type de route (Ex: Urbain, Rural, Autoroute) :  Route rurale
Heure du jour (Ex: Jour, Nuit) :  Nuit
Densité du trafic (Ex: Faible, Moyenne, Élevée) :  Densite moderee
Limitation de vitesse (Ex: 50, 90, 130) :  120
Nombre de véhicules impliqués :  3
Conducteur alcoolisé (Oui/Non) :  Non
```

```
Gravité (Ex: Léger, Grave, Mortel) :  Moderee
État de la route (Ex: Bonne, Dégradée) :  Humide
Type de véhicule (Ex: Voiture, Moto, Camion) :  Camion
Âge du conducteur :  49
Années d'expérience du conducteur :  43
Éclairage de la route (Ex: Bon, Faible, Aucun) :  Lumiere artificielle


 Probabilité estimée d'accident : 0.4115
 Risque d'accident (seuil 0.32) : Oui
```

Conclusion partielle Regression Logistique Le modèle de prédiction présente un taux de mauvais classement autour de 35%, impliquant une marge importante d'erreur. L'analyse des résidus met en lumière une bonne modélisation pour 100% des observations au sens des résidus studentisés, mais suggère qu'une optimisation du modèle est possible. Le seuil optimal pour prédire la survenue d'accident a été déterminé selon l'indice de Youden à 0.32, facilitant ainsi la classification binaire des cas

6 Conclusion générale

Cette étude met en évidence que parmi l'ensemble des variables étudiées, l'état de la route apparaît comme le facteur le plus marqué influençant la probabilité d'accident, notamment les routes sèches par rapport aux routes en construction. Bien que les autres facteurs classiques tels que la météo, l'heure ou la consommation d'alcool n'aient pas été statistiquement associés dans cette analyse, leur rôle ne peut être totalement exclu et pourrait nécessiter des données plus fines ou une stratification plus poussée.

L'efficacité modérée du modèle de régression invite à approfondir cette recherche par l'intégration de variables supplémentaires comme des mesures temporelles fines (conditions météorologiques en temps réel, fatigue du conducteur), des données géospatiales ou encore des comportements dynamiques des conducteurs captés via des technologies embarquées.

Une étude prolongée pourrait également explorer les interactions complexes entre facteurs, l'effet des infrastructures routières plus détaillées (signalisation, virages dangereux), ou encore l'impact de politiques spécifiques mises en œuvre pour améliorer la sécurité routière