Group K

Bayer is launching a new therapy across multiple European markets where pricing rules, reimbursement decisions, and approval timelines vary by country and payer. These market access conditions directly influence how quickly a drug reaches patients and how much market share and revenue it can capture.

This analysis examines which factors Bayer should prioritise across European markets in order to maximise market share and annual sales.

```
import pandas as pd
```

```
df = pd.read_csv("module_C_market_access.csv")
```

```
df.head()
```

	drug_name	country	payer	year	launch_year	launched	policy_strictness	gdp_index	list_price	negotiated_price	reimb
0	Medica17	Austria	ÖGK	2022	2022	1	0.85	0.93	2933	2884.0	
1	Medica05	Sweden	TLV	2024	2022	1	0.90	0.97	2122	2062.0	
2	Medica28	Portugal	SNS	2018	2020	0	0.65	0.70	5364	4829.0	
3	Medica13	France	CNAM	2020	2020	1	0.85	0.95	3243	3121.0	
4	Medica29	Italy	AIFA	2023	2020	1	0.75	0.85	9215	8716.0	

Phase 1: Cleaning the dataset

At this stage, we create a clean working copy of the dataset and address only one clear structural issue.

Some observations do not have a negotiated price recorded. In these cases, we assume that no price negotiation has occurred and set the negotiated price equal to the official list price. This avoids introducing missing values while remaining consistent with real-world pricing logic.Missing values were reviewed and interpreted based on business logic rather than filled or dropped globally.

Note: Outliers were previously examined on r script

```
df.isna().sum().sort_values(ascending=False)
```

	0
reimbursement_delay_months	2891
negotiated_price	790
market_share	511
annual_sales_million	374
drug_name	0
country	0
payer	0
policy_strictness	0
launched	0
launch_year	0
year	0
reimbursed	0
list_price	0
gdp_index	0
access_score	0

dtype: int64

```
df_clean = df.copy()
```

```
df_clean["negotiated_price"] = df_clean["negotiated_price"].fillna(
    df_clean["list_price"]
)
df = df_clean.copy()
```

```
df["negotiated_price"].isna().sum()
```

```
np.int64(0)
```

```
df_clean.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10000 entries, 0 to 9999
Data columns (total 15 columns):
 #   Column                     Non-Null Count  Dtype
---  ------                     --------------  -----
 0   drug_name                  10000 non-null  object
 1   country                    10000 non-null  object
 2   payer                      10000 non-null  object
 3   year                       10000 non-null  int64
 4   launch_year                10000 non-null  int64
 5   launched                   10000 non-null  int64
 6   policy_strictness          10000 non-null  float64
 7   gdp_index                  10000 non-null  float64
 8   list_price                 10000 non-null  int64
 9   negotiated_price           10000 non-null  float64
 10  reimbursed                 10000 non-null  int64
 11  reimbursement_delay_months  7109 non-null   float64
 12  access_score               10000 non-null  float64
 13  market_share               9489 non-null   float64
 14  annual_sales_million       9626 non-null   float64
dtypes: float64(7), int64(5), object(3)
memory usage: 1.1+ MB
```

```
# Dataset used to analyse reimbursement decisions
df_reimb = df_clean.dropna(subset=["reimbursed"])
```

Phase 2 – Reimbursement Classification

In this phase, we analyse which market and policy factors are associated with reimbursement approval.

```
# Inspect the distribution of the reimbursement outcome
df_reimb["reimbursed"].value_counts(normalize=True)
```

reimbursed	proportion
1	0.8112
0	0.1888

dtype: float64

```
# Reimbursement rate by country (top 10 by sample size)
reimb_by_country = (
    df_reimb
    .groupby("country")["reimbursed"]
    .mean()
    .sort_values(ascending=False)
)

reimb_by_country.head(10)
```

country	reimbursed
Germany	0.882497
Denmark	0.874279
Sweden	0.870558
Netherlands	0.859671
Austria	0.840614
France	0.836449
Finland	0.834550
Italy	0.797439
Belgium	0.793417
Spain	0.771868

dtype: float64

```
# Reimbursement rate by GDP quartile
df_reimb["gdp_quartile"] = pd.qcut(df_reimb["gdp_index"], 4, labels=False)

reimb_by_gdp = (
    df_reimb
    .groupby("gdp_quartile")["reimbursed"]
    .mean()
    .reset_index()
)

reimb_by_gdp
```

	gdp_quartile	reimbursed
0	0	0.736716
1	1	0.816551
2	2	0.848723
3	3	0.871035

Exploratory analysis indicates systematic variation in reimbursement, motivating a simple classification model.

Reimbursement Classification Model

A classification model is used to examine which factors are associated with reimbursement approval. The model focuses exclusively on variables that are observable at the time of the reimbursement decision, including country economic conditions, policy environment, and proposed pricing.

Downstream commercial outcomes such as market share and annual sales are intentionally excluded, as these occur after reimbursement and launch. This avoids data leakage and ensures that the model reflects a realistic decision-making context for payers.

```
X = df_reimb[
    [
        "gdp_index",
        "policy_strictness",
        "negotiated_price"
    ]
]

y = df_reimb["reimbursed"]
```

```
from sklearn.tree import DecisionTreeClassifier

# Fit a shallow decision tree for interpretability
tree = DecisionTreeClassifier(
    max_depth=3,
    min_samples_leaf=50,
    random_state=42
```

```
)

tree.fit(X, y)
```

```
DecisionTreeClassifier(max_depth=3, min_samples_leaf=50, random_state=42)
```

```
feature_importance = pd.Series(tree.feature_importances_, index=X.columns).sort_values(ascending=False)
feature_importance
```

	0
policy_strictness	0.854274
negotiated_price	0.145726
gdp_index	0.000000

dtype: float64

Reimbursement decisions are primarily driven by policy strictness, with pricing playing a secondary role once policy conditions are considered.

```
from sklearn.tree import export_text

tree_rules = export_text(tree, feature_names=list(X.columns))
print(tree_rules)
```

```
|--- policy_strictness <= 0.67
|   |--- negotiated_price <= 3291.50
|   |   |--- negotiated_price <= 2449.50
|   |   |   |--- class: 1
|   |   |--- negotiated_price >  2449.50
|   |   |   |--- class: 1
|   |--- negotiated_price >  3291.50
|   |   |--- negotiated_price <= 4126.50
|   |   |   |--- class: 1
|   |   |--- negotiated_price >  4126.50
|   |   |   |--- class: 1
|--- policy_strictness >  0.67
|   |--- policy_strictness <= 0.83
|   |   |--- negotiated_price <= 2732.50
|   |   |   |--- class: 1
|   |   |--- negotiated_price >  2732.50
|   |   |   |--- class: 1
|   |--- policy_strictness >  0.83
|   |   |--- negotiated_price <= 2448.50
|   |   |   |--- class: 1
|   |   |--- negotiated_price >  2448.50
|   |   |   |--- class: 1
```

```
from sklearn.model_selection import train_test_split

X_train, X_test, y_train, y_test = train_test_split(
    X, y,
    test_size=0.3,
    random_state=42,
    stratify=y
)
```

```
tree = DecisionTreeClassifier(
    max_depth=3,
    min_samples_leaf=50,
    random_state=42
)

tree.fit(X_train, y_train)
```

```
DecisionTreeClassifier(max_depth=3, min_samples_leaf=50, random_state=42)
```

```
from sklearn.metrics import classification_report, confusion_matrix

y_pred = tree.predict(X_test)

print(confusion_matrix(y_test, y_pred))
print(classification_report(y_test, y_pred))
```

```
[[   0  566]
 [   0 2434]]
              precision    recall  f1-score   support

           0       0.00      0.00      0.00       566
           1       0.81      1.00      0.90      2434

    accuracy                           0.81      3000
   macro avg       0.41      0.50      0.45      3000
weighted avg       0.66      0.81      0.73      3000

/usr/local/lib/python3.12/dist-packages/sklearn/metrics/_classification.py:1565: UndefinedMetricWarning: Precision is ill-de
  _warn_prf(average, modifier, f"{metric.capitalize()} is", len(result))
/usr/local/lib/python3.12/dist-packages/sklearn/metrics/_classification.py:1565: UndefinedMetricWarning: Precision is ill-de
  _warn_prf(average, modifier, f"{metric.capitalize()} is", len(result))
/usr/local/lib/python3.12/dist-packages/sklearn/metrics/_classification.py:1565: UndefinedMetricWarning: Precision is ill-de
  _warn_prf(average, modifier, f"{metric.capitalize()} is", len(result))
```

Classification Summary

Reimbursement classification was used to explore the main drivers of approval decisions. Policy strictness emerged as the dominant factor, while pricing played a secondary role and country wealth added limited additional explanatory power. Due to high overall reimbursement rates, predictive performance is constrained by class imbalance, so results are interpreted as explanatory rather than predictive.

Phase 3 – Market Share Regression

This phase analyses which market access and pricing factors drive commercial performance once a drug is reimbursed and launched.

Model A — Market Share

Goal: Understand what drives `market_share` across markets once the drug is available.

```
df_m1 = df[df["launched"] == 1].copy()

# Quick check
df_m1.shape
```

```
(7069, 15)
```

```
# Select only the variables we plan to use in Model 1
cols_model1 = [
    "market_share",
    "reimbursement_delay_months",
    "access_score",
    "policy_strictness"
]

df_m1[cols_model1].describe()
```

	market_share	reimbursement_delay_months	access_score	policy_strictness
count	6713.000000	5299.000000	7069.000000	7069.000000
mean	0.100184	3.628156	83.107582	0.802809
std	0.043814	3.049500	21.671760	0.097855
min	0.000000	1.000000	8.800000	0.600000
25%	0.068000	1.000000	82.800000	0.750000
50%	0.102000	2.600000	90.700000	0.850000
75%	0.131000	5.500000	96.800000	0.880000
max	0.256000	18.400000	100.000000	0.900000

```
# Check missing values in the Model 1 variables
df_m1[cols_model1].isna().sum()
```

	0
market_share	356
reimbursement_delay_months	1770
access_score	0
policy_strictness	0

dtype: int64

```
# Create a filled version of reimbursement delay
df_m1["reimbursement_delay_filled"] = df_m1["reimbursement_delay_months"].fillna(999)
```

Missing Values

`market_share` has some missing values, which are left unchanged and will be dropped automatically in the regression.

`reimbursement_delay_months` is missing when the drug is not reimbursed. These are structural missing values, so we encode them as a very large delay to represent non- or very late reimbursement.

```
import statsmodels.formula.api as smf

model_1 = smf.ols(
    "market_share ~ reimbursement_delay_filled + access_score + policy_strictness",
    data=df_m1
).fit()

print(model_1.summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:           market_share   R-squared:                       0.373
Model:                           OLS   Adj. R-squared:                  0.373
Method:                Least Squares   F-statistic:                     1332.
Date:               Mon, 02 Feb 2026   Prob (F-statistic):               0.00
Time:                       19:28:26   Log-Likelihood:                 13041.
No. Observations:               6713   AIC:                        -2.607e+04
Df Residuals:                   6709   BIC:                        -2.605e+04
Df Model:                          3
Covariance Type:            nonrobust
================================================================================================
                                   coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------------------------
Intercept                       0.0003      0.004      0.070      0.944      -0.008       0.008
reimbursement_delay_filled -2.381e-06   1.34e-06     -1.780      0.075      -5e-06    2.42e-07
access_score                    0.0012    2.7e-05     44.360      0.000       0.001       0.001
policy_strictness               0.0014      0.004      0.312      0.755      -0.007       0.010
==============================================================================
Omnibus:                       35.500   Durbin-Watson:                   1.976
Prob(Omnibus):                  0.000   Jarque-Bera (JB):               47.006
Skew:                           0.069   Prob(JB):                     6.21e-11
Kurtosis:                       3.386   Cond. No.                     6.70e+03
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The condition number is large, 6.7e+03. This might indicate that there are
```

```
strong multicollinearity or other numerical problems.
```

Model 1 — Baseline Market Share Model

The baseline model shows that access quality has a strong positive effect on market share, while longer reimbursement delays reduce adoption. Policy strictness does not appear to significantly influence market share on its own.

```
model_2 = smf.ols(
    "market_share ~ reimbursement_delay_filled + access_score + policy_strictness + negotiated_price",
    data=df_m1
).fit()

print(model_2.summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:           market_share   R-squared:                       0.373
Model:                           OLS   Adj. R-squared:                  0.373
Method:                Least Squares   F-statistic:                     999.4
Date:               Mon, 02 Feb 2026   Prob (F-statistic):               0.00
Time:                       19:28:26   Log-Likelihood:                 13041.
No. Observations:               6713   AIC:                         -2.607e+04
Df Residuals:                   6708   BIC:                         -2.604e+04
Df Model:                          4
Covariance Type:            nonrobust
================================================================================================
                                   coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------------------------
Intercept                      -0.0001      0.004     -0.026      0.979      -0.008       0.008
reimbursement_delay_filled -2.399e-06   1.34e-06     -1.793      0.073   -5.02e-06    2.24e-07
access_score                    0.0012    2.7e-05     44.330      0.000       0.001       0.001
policy_strictness               0.0012      0.004      0.271      0.787      -0.008       0.010
negotiated_price             1.037e-07   1.39e-07      0.745      0.456   -1.69e-07    3.76e-07
==============================================================================
Omnibus:                       35.397   Durbin-Watson:                   1.977
Prob(Omnibus):                  0.000   Jarque-Bera (JB):               46.779
Skew:                           0.070   Prob(JB):                     6.95e-11
Kurtosis:                       3.384   Cond. No.                     8.45e+04
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The condition number is large, 8.45e+04. This might indicate that there are
strong multicollinearity or other numerical problems.
```

Model 2 — Market Share with Pricing Control

After adding negotiated price, the results remain unchanged. Access quality continues to be the main driver of market share, while pricing does not significantly affect adoption. This suggests that access conditions matter more than price for market penetration.

```
model_3 = smf.ols(
    "market_share ~ reimbursement_delay_filled + access_score + negotiated_price + C(country)",
    data=df_m1
).fit()

print(model_3.summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:           market_share   R-squared:                       0.374
Model:                           OLS   Adj. R-squared:                  0.373
Method:                Least Squares   F-statistic:                     285.8
Date:               Mon, 02 Feb 2026   Prob (F-statistic):               0.00
Time:                       19:28:26   Log-Likelihood:                 13044.
No. Observations:               6713   AIC:                         -2.606e+04
Df Residuals:                   6698   BIC:                         -2.596e+04
Df Model:                         14
Covariance Type:            nonrobust
================================================================================================
                                   coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------------------------
Intercept                      -0.0004      0.003     -0.139      0.889      -0.006       0.005
C(country)[T.Belgium]           0.0013      0.002      0.623      0.533      -0.003       0.005
C(country)[T.Denmark]           0.0018      0.002      0.870      0.385      -0.002       0.006
C(country)[T.Finland]          -0.0007      0.002     -0.317      0.751      -0.005       0.003
C(country)[T.France]            0.0015      0.002      0.714      0.475      -0.003       0.005
C(country)[T.Germany]           0.0025      0.002      1.228      0.220      -0.001       0.006
C(country)[T.Italy]             0.0032      0.002      1.583      0.113      -0.001       0.007
C(country)[T.Netherlands]       0.0011      0.002      0.503      0.615      -0.003       0.005
C(country)[T.Poland]            0.0008      0.002      0.403      0.687      -0.003       0.005
```

```
C(country)[T.Portugal]         0.0006      0.002      0.286      0.775      -0.004       0.005
C(country)[T.Spain]            0.0004      0.002      0.176      0.860      -0.004       0.004
C(country)[T.Sweden]           0.0013      0.002      0.626      0.531      -0.003       0.005
reimbursement_delay_filled -2.324e-06   1.34e-06     -1.734      0.083   -4.95e-06    3.04e-07
access_score                   0.0012   2.71e-05     44.268      0.000       0.001       0.001
negotiated_price           1.076e-07   1.39e-07      0.773      0.440   -1.65e-07    3.81e-07
==============================================================================
Omnibus:                       35.526   Durbin-Watson:                   1.976
Prob(Omnibus):                  0.000   Jarque-Bera (JB):               46.961
Skew:                           0.070   Prob(JB):                     6.35e-11
Kurtosis:                       3.385   Cond. No.                     7.87e+04
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The condition number is large, 7.87e+04. This might indicate that there are
strong multicollinearity or other numerical problems.
```

Model 3 — Market Share with Country Fixed Effects

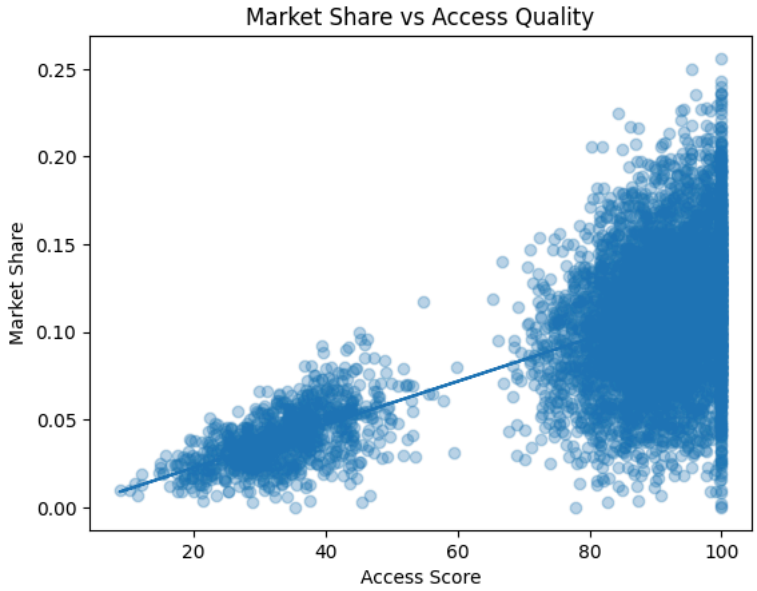
Including country fixed effects does not materially alter the results. Access quality remains the dominant determinant of market share, and reimbursement delays continue to have a negative impact. The findings are robust across European markets.

```
model_3_final = smf.ols(
    "market_share ~ access_score + reimbursement_delay_filled + C(country)",
    data=df_m1
).fit()

print(model_3_final.summary())
```

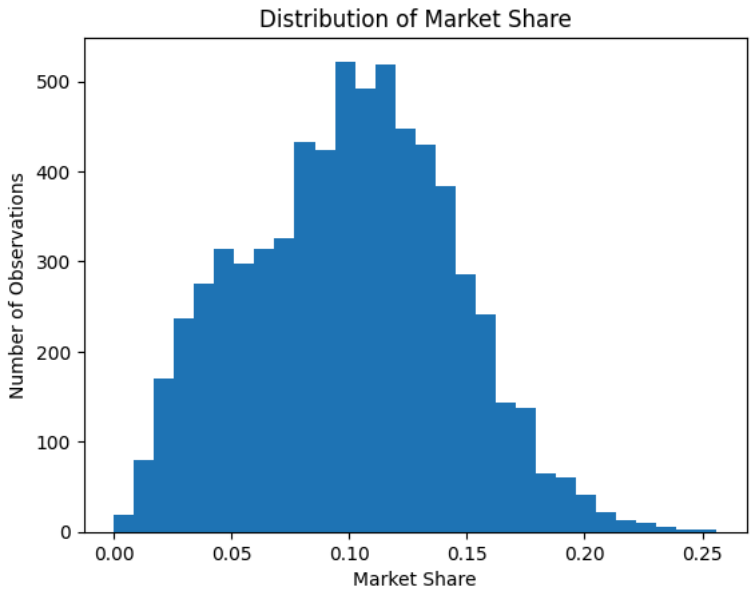
```
                            OLS Regression Results
==============================================================================
Dep. Variable:           market_share   R-squared:                       0.374
Model:                           OLS   Adj. R-squared:                  0.373
Method:                Least Squares   F-statistic:                     307.7
Date:               Mon, 02 Feb 2026   Prob (F-statistic):               0.00
Time:                       19:28:26   Log-Likelihood:                 13044.
No. Observations:               6713   AIC:                         -2.606e+04
Df Residuals:                   6699   BIC:                         -2.596e+04
Df Model:                         13
Covariance Type:           nonrobust
================================================================================================
                                   coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------------------------
Intercept                        0.0002      0.003      0.055      0.956      -0.006       0.006
C(country)[T.Belgium]            0.0012      0.002      0.606      0.545      -0.003       0.005
C(country)[T.Denmark]            0.0018      0.002      0.865      0.387      -0.002       0.006
C(country)[T.Finland]           -0.0007      0.002     -0.319      0.750      -0.005       0.003
C(country)[T.France]             0.0014      0.002      0.707      0.480      -0.003       0.005
C(country)[T.Germany]            0.0025      0.002      1.226      0.220      -0.001       0.006
C(country)[T.Italy]              0.0032      0.002      1.562      0.118      -0.001       0.007
C(country)[T.Netherlands]        0.0010      0.002      0.488      0.626      -0.003       0.005
C(country)[T.Poland]             0.0008      0.002      0.371      0.711      -0.003       0.005
C(country)[T.Portugal]           0.0006      0.002      0.263      0.793      -0.004       0.005
C(country)[T.Spain]              0.0003      0.002      0.149      0.881      -0.004       0.004
C(country)[T.Sweden]             0.0013      0.002      0.613      0.540      -0.003       0.005
access_score                     0.0012    2.7e-05     44.299      0.000       0.001       0.001
reimbursement_delay_filled -2.306e-06   1.34e-06     -1.720      0.085   -4.93e-06    3.22e-07
==============================================================================
Omnibus:                       35.640   Durbin-Watson:                   1.976
Prob(Omnibus):                  0.000   Jarque-Bera (JB):               47.209
Skew:                           0.069   Prob(JB):                     5.61e-11
Kurtosis:                       3.387   Cond. No.                     6.22e+03
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The condition number is large, 6.22e+03. This might indicate that there are
strong multicollinearity or other numerical problems.
```

```
import matplotlib.pyplot as plt
import numpy as np

# Drop rows with missing values for the plot
plot_df = df_m1[["access_score", "market_share"]].dropna()

# Scatter plot
plt.scatter(plot_df["access_score"], plot_df["market_share"], alpha=0.3)

# Fit and plot regression line
m, b = np.polyfit(plot_df["access_score"], plot_df["market_share"], 1)
plt.plot(plot_df["access_score"], m * plot_df["access_score"] + b)
```

```
plt.xlabel("Access Score")
plt.ylabel("Market Share")
plt.title("Market Share vs Access Quality")
plt.show()
```




```
plt.hist(df_m1["market_share"].dropna(), bins=30)
plt.xlabel("Market Share")
plt.ylabel("Number of Observations")
plt.title("Distribution of Market Share")
plt.show()
```




```
from statsmodels.stats.outliers_influence import variance_inflation_factor
import statsmodels.api as sm

vif_vars = df_m1[[
    "access_score",
    "reimbursement_delay_filled",
    "negotiated_price"
]].dropna()

X = sm.add_constant(vif_vars)

vif_df = pd.DataFrame()
vif_df["Variable"] = X.columns
vif_df["VIF"] = [variance_inflation_factor(X.values, i) for i in range(X.shape[1])]

vif_df
```

	Variable	VIF
0	const	37.229784
1	access_score	1.855343
2	reimbursement_delay_filled	1.854187
3	negotiated_price	1.001015

Summary — Market Share Analysis

Across multiple model specifications, market share is primarily driven by market access conditions rather than pricing. Higher access quality consistently leads to greater adoption, while longer reimbursement delays reduce market share. These results are robust to the inclusion of pricing controls and country fixed effects.

Having established how access conditions shape adoption, we now turn to annual sales to understand how market share, pricing, and market size interact to determine revenue.

Model B — Annual Sales (Revenue Model): Analysis Plan

Having analysed what drives market share, we now examine what determines annual sales

```
# 1) Missing values in annual sales
df["annual_sales_million"].isna().sum()
```

```
np.int64(374)
```

```
df.groupby("launched")["annual_sales_million"].describe()
```

launched	count	mean	std	min	25%	50%	75%	max
0	2831.0	0.000000	0.000000	0.0	0.00	0.00	0.00	0.00
1	6795.0	0.067363	0.055341	0.0	0.03	0.05	0.09	0.66

For the sales analysis, we restrict the sample to observations where the drug is launched. When `launched = 0`, annual sales are mechanically zero and contain no informative variation. The revenue model therefore focuses only on launched observations.

```
# Dataset for sales model
df_sales = df[df["launched"] == 1].copy()

df_sales.shape
```

```
(7069, 15)
```

```
# Add the filled reimbursement delay column to the sales dataset
df_sales["reimbursement_delay_filled"] = df_sales["reimbursement_delay_months"].fillna(999)
```

```
df_sales["annual_sales_million"].isna().mean()
```

```
np.float64(0.03876078653274862)
```

```
import statsmodels.formula.api as smf

model_4 = smf.ols(
    """
    annual_sales_million ~
    market_share
    + negotiated_price
    + gdp_index
    + access_score
    + reimbursement_delay_filled
    + policy_strictness
    + C(country)
    """,
    data=df_sales
```

```
).fit()

print(model_4.summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:     annual_sales_million   R-squared:                       0.767
Model:                              OLS   Adj. R-squared:                  0.766
Method:                   Least Squares   F-statistic:                     1411.
Date:                  Mon, 02 Feb 2026   Prob (F-statistic):               0.00
Time:                          19:28:28   Log-Likelihood:                 14293.
No. Observations:                  6458   AIC:                         -2.855e+04
Df Residuals:                      6442   BIC:                         -2.845e+04
Df Model:                            15
Covariance Type:              nonrobust
================================================================================================
                                   coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------------------------
Intercept                       -0.0301      0.001    -32.738      0.000      -0.032      -0.028
C(country)[T.Belgium]           -0.0017      0.002     -1.131      0.258      -0.005       0.001
C(country)[T.Denmark]            0.0034      0.002      2.126      0.034       0.000       0.007
C(country)[T.Finland]            0.0010      0.002      0.598      0.550      -0.002       0.004
C(country)[T.France]            -0.0003      0.002     -0.202      0.840      -0.003       0.003
C(country)[T.Germany]            0.0024      0.002      1.491      0.136      -0.001       0.006
C(country)[T.Italy]             -0.0049      0.001     -3.246      0.001      -0.008      -0.002
C(country)[T.Netherlands]        0.0012      0.002      0.747      0.455      -0.002       0.004
C(country)[T.Poland]            -0.0108      0.001     -7.752      0.000      -0.014      -0.008
C(country)[T.Portugal]          -0.0107      0.001     -7.363      0.000      -0.013      -0.008
C(country)[T.Spain]             -0.0045      0.001     -3.028      0.002      -0.007      -0.002
C(country)[T.Sweden]             0.0034      0.002      2.058      0.040       0.000       0.007
market_share                     0.6571      0.009     69.250      0.000       0.639       0.676
negotiated_price            1.235e-05   1.09e-07    113.767      0.000    1.21e-05    1.26e-05
gdp_index                       -0.0211      0.001    -24.077      0.000      -0.023      -0.019
access_score                -1.198e-05   2.38e-05     -0.503      0.615   -5.87e-05    3.48e-05
reimbursement_delay_filled -4.791e-07   1.04e-06     -0.462      0.644   -2.51e-06    1.55e-06
policy_strictness               -0.0190      0.001    -21.544      0.000      -0.021      -0.017
==============================================================================
Omnibus:                     2371.962   Durbin-Watson:                   2.014
Prob(Omnibus):                  0.000   Jarque-Bera (JB):            44389.911
Skew:                           1.278   Prob(JB):                         0.00
Kurtosis:                      15.587   Cond. No.                     4.17e+19
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The smallest eigenvalue is 1.48e-28. This might indicate that there are
strong multicollinearity problems or that the design matrix is singular.
```

Insights from Initial Sales Model

The first sales regression shows that annual sales are strongly driven by market share and negotiated price. Market share has a large and highly significant positive coefficient, confirming that adoption is the primary driver of revenue. Negotiated price is also strongly positive and significant, reflecting that higher effective prices directly increase sales once access is achieved. GDP captures market size effects, while access variables such as access score and reimbursement delay are no longer significant once market share is included, indicating that their impact on sales operates indirectly through adoption.

```
from statsmodels.stats.outliers_influence import variance_inflation_factor
import statsmodels.api as sm

vif_vars_sales = df_sales[[
    "market_share",
    "negotiated_price",
    "gdp_index",
    "access_score",
    "reimbursement_delay_filled",
    "policy_strictness"
]].dropna()

X = sm.add_constant(vif_vars_sales)

vif_df = pd.DataFrame()
vif_df["Variable"] = X.columns
vif_df["VIF"] = [
    variance_inflation_factor(X.values, i)
    for i in range(X.shape[1])
]

vif_df
```

	Variable	VIF
0	const	93.296879
1	market_share	1.596093
2	negotiated_price	1.004369
3	gdp_index	23.842230
4	access_score	2.486578
5	reimbursement_delay_filled	1.864748
6	policy_strictness	23.834811

Insights from Multicollinearity (VIF) Analysis

The VIF analysis reveals no multicollinearity concerns among market share, negotiated price, access score, or reimbursement delay. However, GDP index and policy strictness exhibit very high VIF values due to strong overlap with country fixed effects, as both variables are largely country-level characteristics. This indicates redundancy rather than model misspecification and motivates removing policy strictness from the final sales model while retaining GDP to capture market size.

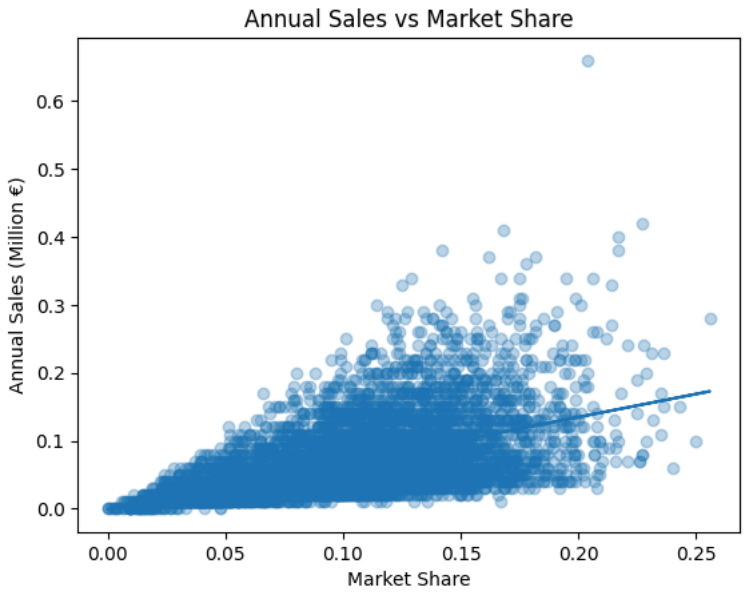
```
model_4_final = smf.ols(
    """
    annual_sales_million ~
    market_share
    + negotiated_price
    + gdp_index
    + access_score
    + reimbursement_delay_filled
    + C(country)
    """,
    data=df_sales
).fit()

print(model_4_final.summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:     annual_sales_million   R-squared:                       0.767
Model:                              OLS   Adj. R-squared:                  0.766
Method:                   Least Squares   F-statistic:                     1411.
Date:                  Mon, 02 Feb 2026   Prob (F-statistic):               0.00
Time:                          19:28:29   Log-Likelihood:                 14293.
No. Observations:                  6458   AIC:                         -2.855e+04
Df Residuals:                      6442   BIC:                         -2.845e+04
Df Model:                            15
Covariance Type:              nonrobust
================================================================================================
                                   coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------------------------
Intercept                      -0.0381      0.001    -31.358      0.000      -0.040      -0.036
C(country)[T.Belgium]          -0.0009      0.002     -0.562      0.574      -0.004       0.002
C(country)[T.Denmark]           0.0031      0.002      1.944      0.052   -2.62e-05       0.006
C(country)[T.Finland]           0.0007      0.002      0.418      0.676      -0.002       0.004
C(country)[T.France]           -0.0001      0.002     -0.091      0.927      -0.003       0.003
C(country)[T.Germany]           0.0021      0.002      1.293      0.196      -0.001       0.005
C(country)[T.Italy]            -0.0037      0.002     -2.404      0.016      -0.007      -0.001
C(country)[T.Netherlands]       0.0011      0.002      0.670      0.503      -0.002       0.004
C(country)[T.Poland]           -0.0086      0.001     -5.932      0.000      -0.011      -0.006
C(country)[T.Portugal]         -0.0089      0.001     -5.986      0.000      -0.012      -0.006
C(country)[T.Spain]            -0.0026      0.002     -1.714      0.087      -0.006       0.000
C(country)[T.Sweden]            0.0028      0.002      1.709      0.088      -0.000       0.006
market_share                    0.6571      0.009     69.250      0.000       0.639       0.676
negotiated_price             1.235e-05   1.09e-07    113.767      0.000    1.21e-05    1.26e-05
gdp_index                      -0.0300      0.001    -23.334      0.000      -0.032      -0.027
access_score                -1.198e-05   2.38e-05     -0.503      0.615   -5.87e-05    3.48e-05
reimbursement_delay_filled  -4.791e-07   1.04e-06     -0.462      0.644   -2.51e-06    1.55e-06
==============================================================================
Omnibus:                     2371.962   Durbin-Watson:                   2.014
Prob(Omnibus):                  0.000   Jarque-Bera (JB):            44389.911
Skew:                           1.278   Prob(JB):                         0.00
Kurtosis:                      15.587   Cond. No.                     5.55e+18
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The smallest eigenvalue is 8.36e-27. This might indicate that there are
strong multicollinearity problems or that the design matrix is singular.
```

Insights from Final Sales Model

After removing policy strictness, the final sales model confirms that revenue is primarily determined by three factors: market share, negotiated price, and market size. Market share remains the dominant driver of annual sales, highlighting the importance of adoption. Negotiated price has a strong positive effect, showing that pricing decisions materially affect revenue once access is secured. GDP index captures differences in market scale across countries. Access score and reimbursement delay are not directly significant in this model, reinforcing the conclusion that access conditions influence sales indirectly through their effect on market share rather than directly.

```
import matplotlib.pyplot as plt
import numpy as np

plot_df = df_sales[["market_share", "annual_sales_million"]].dropna()

plt.scatter(plot_df["market_share"], plot_df["annual_sales_million"], alpha=0.3)

m, b = np.polyfit(plot_df["market_share"], plot_df["annual_sales_million"], 1)
plt.plot(plot_df["market_share"], m * plot_df["market_share"] + b)

plt.xlabel("Market Share")
plt.ylabel("Annual Sales (Million €)")
plt.title("Annual Sales vs Market Share")
plt.show()
```



Phase 4: Clustering Analysis

To complement the regression analysis, we apply an unsupervised clustering approach to identify common market archetypes across European countries and payers.

```
from sklearn.preprocessing import StandardScaler

# Select clustering variables
cluster_vars = [
    "access_score",
    "reimbursement_delay_filled",
    "market_share",
    "negotiated_price",
    "gdp_index"
]

# Drop rows with missing values
df_cluster = df_sales[cluster_vars].dropna().copy()

# Standardise variables
scaler = StandardScaler()
X_scaled = scaler.fit_transform(df_cluster)

X_scaled.shape
```

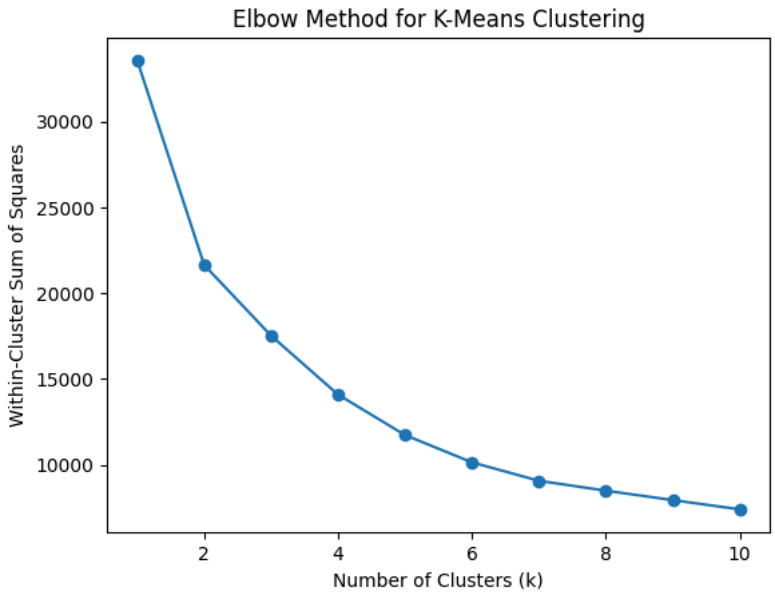
```
(6713, 5)
```

Variable Selection for Clustering

The clustering analysis uses a small set of numerical variables that capture key dimensions of market structure and performance. Access score and reimbursement delay represent market access conditions, market share captures adoption, negotiated price reflects the effective price paid by payers, and GDP index proxies market size.

These variables are chosen to describe how markets differ in terms of access, adoption, pricing, and scale, without directly clustering on revenue outcomes or imposing country-level labels. This allows the clustering to reveal natural market archetypes rather than pre-defined groupings.

```
from sklearn.cluster import KMeans
import matplotlib.pyplot as plt

inertia = []
K_range = range(1, 11)

for k in K_range:
    kmeans = KMeans(n_clusters=k, random_state=42, n_init=10)
    kmeans.fit(X_scaled)
    inertia.append(kmeans.inertia_)

plt.plot(K_range, inertia, marker='o')
plt.xlabel("Number of Clusters (k)")
plt.ylabel("Within-Cluster Sum of Squares")
plt.title("Elbow Method for K-Means Clustering")
plt.show()
```




```
# Fit K-means with k = 3
kmeans = KMeans(n_clusters=3, random_state=42, n_init=10)
clusters = kmeans.fit_predict(X_scaled)

# Add cluster labels back to the data
df_clustered = df_cluster.copy()
df_clustered["cluster"] = clusters

df_clustered["cluster"].value_counts()
```

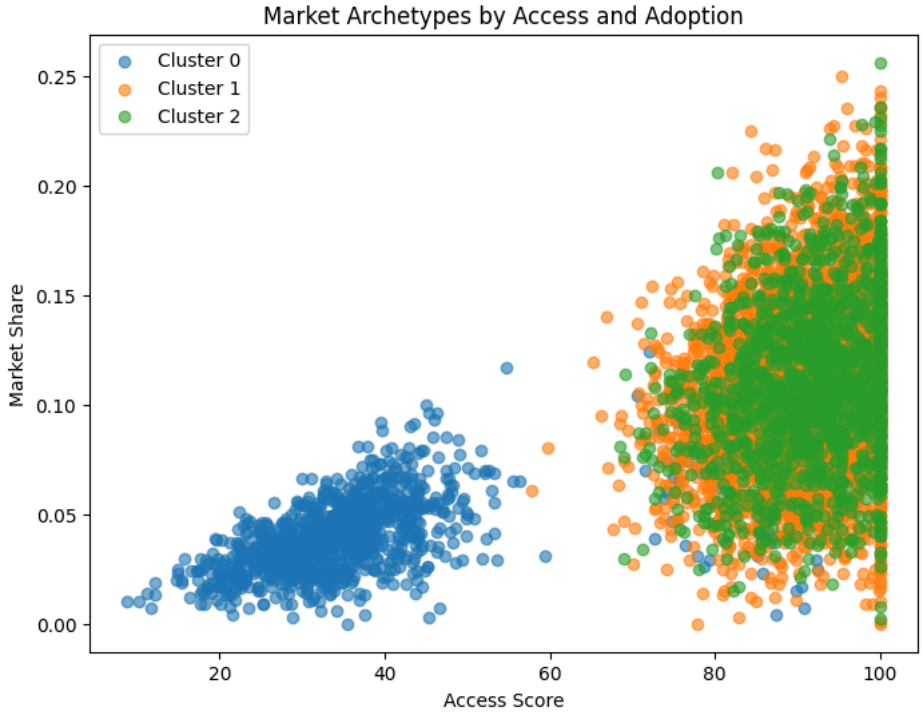
cluster	count
1	4030
2	1624
0	1059

dtype: int64

```
cluster_summary = (
    df_clustered
    .groupby("cluster")
    .mean()
    .round(3)
)

cluster_summary
```

cluster	access_score	reimbursement_delay_filled	market_share	negotiated_price	gdp_index
0	36.909	998.060	0.041	5341.740	0.846
1	91.635	108.856	0.111	3921.834	0.898
2	91.792	129.229	0.111	9681.721	0.901

```
import matplotlib.pyplot as plt

plt.figure(figsize=(8, 6))

for cluster in sorted(df_clustered["cluster"].unique()):
    subset = df_clustered[df_clustered["cluster"] == cluster]
    plt.scatter(
        subset["access_score"],
        subset["market_share"],
        label=f"Cluster {cluster}",
        alpha=0.6
    )

plt.xlabel("Access Score")
plt.ylabel("Market Share")
plt.title("Market Archetypes by Access and Adoption")
plt.legend()
plt.show()
```



Clustering Results (k = 3)

Using the elbow method, a three-cluster solution was initially selected to explore potential market archetypes. One cluster emerges as clearly distinct, characterised by low access scores, the absence of reimbursement, and very low market share.

However, visual inspection of the clustering results shows that the remaining two clusters overlap substantially in access and adoption space. These overlapping clusters both represent access-enabled markets and are primarily differentiated by pricing levels and market size rather than by adoption outcomes. As such, the three-cluster solution is used for exploratory insight into strategic differences within access-enabled markets rather than as a strict market segmentation.

```
kmeans_2 = KMeans(n_clusters=2, random_state=42, n_init=10)
clusters_2 = kmeans_2.fit_predict(X_scaled)

df_clustered_2 = df_cluster.copy()
df_clustered_2["cluster"] = clusters_2

df_clustered_2.groupby("cluster").mean().round(3)
```

cluster	access_score	reimbursement_delay_filled	market_share	negotiated_price	gdp_index
0	91.707	112.041	0.111	5570.197	0.899
1	37.634	998.075	0.041	5377.149	0.846

Clustering Results (k = 2)

In addition to the three-cluster solution used for exploratory insights, we also examine a two-cluster specification for clarity and interpretability. Visual inspection shows one group of markets is clearly distinct, while the remaining markets form a single dense cluster. A two-cluster solution therefore captures the most fundamental structural divide in the data: whether markets are access-constrained or access-enabled.

From Analysis to Managerial Recommendations

Through a combination of classification, regression, and clustering analyses, we have examined market access outcomes, adoption dynamics, revenue drivers, and market archetypes across European markets. Together, these approaches provide a coherent and triangulated view of how access conditions, pricing, and market characteristics interact to shape performance.