nlu_rag_openml_eda

November 11, 2025

1 Agentic ML Builder

1.0.1 The Agentic ML Builder is more of a ML Scafolding code generator targeting improvement of productivity of ML Engineers

The Agentic ML Builder uses NLU Intent Extractor for Transformer model (fine-tuned Azure GPT-4, or smaller DistilBERT) to Convert project description to structured configuration ({task: classification, model: CNN, data: images}).

Uses RAG Template Retriever for Retrieve matching code template or architecture for the project.

1.0.2 This notebook is used to perform EDA NLU and RAG on example datasets

2 NLU and RAG EDA with OpenML datasets

This notebook demonstrates how to fetch datasets and use them as examples for NLU and RAG template selection.

[15]:
```
# Requirements: scikit-learn, pandas, sentence-transformers
! pip install scikit-learn pandas sentence-transformers openml
```

```
Requirement already satisfied: scikit-learn in
d:\03.2025usd\aai-590\proj\code\agenticmlbuilder\agmlb\lib\site-packages (1.7.2)
Requirement already satisfied: pandas in
d:\03.2025usd\aai-590\proj\code\agenticmlbuilder\agmlb\lib\site-packages (2.3.3)
Requirement already satisfied: sentence-transformers in
d:\03.2025usd\aai-590\proj\code\agenticmlbuilder\agmlb\lib\site-packages (5.1.2)
Requirement already satisfied: openml in
d:\03.2025usd\aai-590\proj\code\agenticmlbuilder\agmlb\lib\site-packages
(0.15.1)
Requirement already satisfied: numpy>=1.22.0 in
d:\03.2025usd\aai-590\proj\code\agenticmlbuilder\agmlb\lib\site-packages (from
scikit-learn) (2.3.4)
Requirement already satisfied: scipy>=1.8.0 in
d:\03.2025usd\aai-590\proj\code\agenticmlbuilder\agmlb\lib\site-packages (from
scikit-learn) (1.16.3)
Requirement already satisfied: joblib>=1.2.0 in
d:\03.2025usd\aai-590\proj\code\agenticmlbuilder\agmlb\lib\site-packages (from
scikit-learn) (1.5.2)
```

```
Requirement already satisfied: threadpoolctl>=3.1.0 in
d:\03.2025usd\aai-590\proj\code\agenticmlbuilder\agmlb\lib\site-packages (from
scikit-learn) (3.6.0)
Requirement already satisfied: python-dateutil>=2.8.2 in
d:\03.2025usd\aai-590\proj\code\agenticmlbuilder\agmlb\lib\site-packages (from
pandas) (2.9.0.post0)
Requirement already satisfied: pytz>=2020.1 in
d:\03.2025usd\aai-590\proj\code\agenticmlbuilder\agmlb\lib\site-packages (from
pandas) (2025.2)
Requirement already satisfied: tzdata>=2022.7 in
d:\03.2025usd\aai-590\proj\code\agenticmlbuilder\agmlb\lib\site-packages (from
pandas) (2025.2)
Requirement already satisfied: transformers<5.0.0,>=4.41.0 in
d:\03.2025usd\aai-590\proj\code\agenticmlbuilder\agmlb\lib\site-packages (from
sentence-transformers) (4.57.1)
Requirement already satisfied: tqdm in
d:\03.2025usd\aai-590\proj\code\agenticmlbuilder\agmlb\lib\site-packages (from
sentence-transformers) (4.67.1)
Requirement already satisfied: torch>=1.11.0 in
d:\03.2025usd\aai-590\proj\code\agenticmlbuilder\agmlb\lib\site-packages (from
sentence-transformers) (2.9.0)
Requirement already satisfied: huggingface-hub>=0.20.0 in
d:\03.2025usd\aai-590\proj\code\agenticmlbuilder\agmlb\lib\site-packages (from
sentence-transformers) (0.36.0)
Requirement already satisfied: Pillow in
d:\03.2025usd\aai-590\proj\code\agenticmlbuilder\agmlb\lib\site-packages (from
sentence-transformers) (12.0.0)
Requirement already satisfied: typing_extensions>=4.5.0 in
d:\03.2025usd\aai-590\proj\code\agenticmlbuilder\agmlb\lib\site-packages (from
sentence-transformers) (4.15.0)
Requirement already satisfied: filelock in
d:\03.2025usd\aai-590\proj\code\agenticmlbuilder\agmlb\lib\site-packages (from
transformers<5.0.0,>=4.41.0->sentence-transformers) (3.20.0)
Requirement already satisfied: packaging>=20.0 in
d:\03.2025usd\aai-590\proj\code\agenticmlbuilder\agmlb\lib\site-packages (from
transformers<5.0.0,>=4.41.0->sentence-transformers) (25.0)
Requirement already satisfied: pyyaml>=5.1 in
d:\03.2025usd\aai-590\proj\code\agenticmlbuilder\agmlb\lib\site-packages (from
transformers<5.0.0,>=4.41.0->sentence-transformers) (6.0.3)
Requirement already satisfied: regex!=2019.12.17 in
d:\03.2025usd\aai-590\proj\code\agenticmlbuilder\agmlb\lib\site-packages (from
transformers<5.0.0,>=4.41.0->sentence-transformers) (2025.11.3)
Requirement already satisfied: requests in
d:\03.2025usd\aai-590\proj\code\agenticmlbuilder\agmlb\lib\site-packages (from
transformers<5.0.0,>=4.41.0->sentence-transformers) (2.32.5)
Requirement already satisfied: tokenizers<=0.23.0,>=0.22.0 in
d:\03.2025usd\aai-590\proj\code\agenticmlbuilder\agmlb\lib\site-packages (from
transformers<5.0.0,>=4.41.0->sentence-transformers) (0.22.1)
```

```
Requirement already satisfied: safetensors>=0.4.3 in
d:\03.2025usd\aai-590\proj\code\agenticmlbuilder\agmlb\lib\site-packages (from
transformers<5.0.0,>=4.41.0->sentence-transformers) (0.6.2)
Requirement already satisfied: fsspec>=2023.5.0 in
d:\03.2025usd\aai-590\proj\code\agenticmlbuilder\agmlb\lib\site-packages (from
huggingface-hub>=0.20.0->sentence-transformers) (2025.10.0)
Requirement already satisfied: liac-arff>=2.4.0 in
d:\03.2025usd\aai-590\proj\code\agenticmlbuilder\agmlb\lib\site-packages (from
openml) (2.5.0)
Requirement already satisfied: xmltodict in
d:\03.2025usd\aai-590\proj\code\agenticmlbuilder\agmlb\lib\site-packages (from
openml) (1.0.2)
Requirement already satisfied: minio in
d:\03.2025usd\aai-590\proj\code\agenticmlbuilder\agmlb\lib\site-packages (from
openml) (7.2.18)
Requirement already satisfied: pyarrow in
d:\03.2025usd\aai-590\proj\code\agenticmlbuilder\agmlb\lib\site-packages (from
openml) (21.0.0)
Requirement already satisfied: six>=1.5 in
d:\03.2025usd\aai-590\proj\code\agenticmlbuilder\agmlb\lib\site-packages (from
python-dateutil>=2.8.2->pandas) (1.17.0)
Requirement already satisfied: sympy>=1.13.3 in
d:\03.2025usd\aai-590\proj\code\agenticmlbuilder\agmlb\lib\site-packages (from
torch>=1.11.0->sentence-transformers) (1.14.0)
Requirement already satisfied: networkx>=2.5.1 in
d:\03.2025usd\aai-590\proj\code\agenticmlbuilder\agmlb\lib\site-packages (from
torch>=1.11.0->sentence-transformers) (3.5)
Requirement already satisfied: jinja2 in
d:\03.2025usd\aai-590\proj\code\agenticmlbuilder\agmlb\lib\site-packages (from
torch>=1.11.0->sentence-transformers) (3.1.6)
Requirement already satisfied: setuptools in
d:\03.2025usd\aai-590\proj\code\agenticmlbuilder\agmlb\lib\site-packages (from
torch>=1.11.0->sentence-transformers) (80.9.0)
Requirement already satisfied: mpmath<1.4,>=1.1.0 in
d:\03.2025usd\aai-590\proj\code\agenticmlbuilder\agmlb\lib\site-packages (from
sympy>=1.13.3->torch>=1.11.0->sentence-transformers) (1.3.0)
Requirement already satisfied: colorama in
d:\03.2025usd\aai-590\proj\code\agenticmlbuilder\agmlb\lib\site-packages (from
tqdm->sentence-transformers) (0.4.6)
Requirement already satisfied: MarkupSafe>=2.0 in
d:\03.2025usd\aai-590\proj\code\agenticmlbuilder\agmlb\lib\site-packages (from
jinja2->torch>=1.11.0->sentence-transformers) (3.0.3)
Requirement already satisfied: argon2-cffi in
d:\03.2025usd\aai-590\proj\code\agenticmlbuilder\agmlb\lib\site-packages (from
minio->openml) (25.1.0)
Requirement already satisfied: certifi in
d:\03.2025usd\aai-590\proj\code\agenticmlbuilder\agmlb\lib\site-packages (from
minio->openml) (2025.10.5)
```

```
Requirement already satisfied: pycryptodome in
d:\03.2025usd\aai-590\proj\code\agenticmlbuilder\agmlb\lib\site-packages (from
minio->openml) (3.23.0)
Requirement already satisfied: urllib3 in
d:\03.2025usd\aai-590\proj\code\agenticmlbuilder\agmlb\lib\site-packages (from
minio->openml) (2.5.0)
Requirement already satisfied: argon2-cffi-bindings in
d:\03.2025usd\aai-590\proj\code\agenticmlbuilder\agmlb\lib\site-packages (from
argon2-cffi->minio->openml) (25.1.0)
Requirement already satisfied: cffi>=1.0.1 in
d:\03.2025usd\aai-590\proj\code\agenticmlbuilder\agmlb\lib\site-packages (from
argon2-cffi-bindings->argon2-cffi->minio->openml) (2.0.0)
Requirement already satisfied: pycparser in
d:\03.2025usd\aai-590\proj\code\agenticmlbuilder\agmlb\lib\site-packages (from
cffi>=1.0.1->argon2-cffi-bindings->argon2-cffi->minio->openml) (2.23)
Requirement already satisfied: charset_normalizer<4,>=2 in
d:\03.2025usd\aai-590\proj\code\agenticmlbuilder\agmlb\lib\site-packages (from
requests->transformers<5.0.0,>=4.41.0->sentence-transformers) (3.4.4)
Requirement already satisfied: idna<4,>=2.5 in
d:\03.2025usd\aai-590\proj\code\agenticmlbuilder\agmlb\lib\site-packages (from
requests->transformers<5.0.0,>=4.41.0->sentence-transformers) (3.11)
```

[16]:
```python
import openml
import pandas as pd
from sentence_transformers import SentenceTransformer

# Example: load local CSVs saved by notebook 1
datasets = ['iris_openml.csv','wine_openml.csv','diabetes_openml.csv']
for p in datasets:
    try:
        df = pd.read_csv(p)
        print(p, '->', df.shape)
    except Exception as e:
        print('Missing', p)

# Build sample RAG corpus from template descriptions
templates = [
    {'template_id':'tpl_iris','task_type':'classification','description':'Small␣
 ↪iris classification scikit-learn template'},
    {'template_id':'tpl_wine','task_type':'classification','description':'Wine␣
 ↪quality prediction template'},
    {'template_id':'tpl_diabetes','task_type':'regression','description':
 ↪'Diabetes regression example'}
]
model = SentenceTransformer('all-MiniLM-L6-v2')
corpus = [t['description'] for t in templates]
embeddings = model.encode(corpus)
```

```
print('Built sample embeddings for RAG corpus')
```

```
iris_openml.csv -> (150, 6)
wine_openml.csv -> (178, 15)
diabetes_openml.csv -> (768, 10)
Built sample embeddings for RAG corpus
```

[17]:
```
# EDA Notebook-ready script for Jupyter
# Place this code into a single Jupyter cell or split into cells as indicated␣
 ↪by the comments.

# %%
# Cell 1 - Imports and configuration
import os
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from matplotlib.ticker import MaxNLocator
from sklearn.decomposition import PCA
from sklearn.manifold import TSNE
from sklearn.preprocessing import StandardScaler
from IPython.display import display, Markdown

# Optional: increase inline figure size default for the notebook
plt.rcParams['figure.figsize'] = (8, 5)

# %%
# Cell 2 - File paths (edit if needed)
DATA_FILES = {
    'iris': 'iris_openml.csv',
    'wine': 'wine_openml.csv',
    'diabetes': 'diabetes_openml.csv',
}
EMBEDDING_PATHS = [
    'embeddings.npy',
    'embeddings.csv',
]

# %%
# Cell 3 - Utilities for loading and reporting

def safe_read_csv(path):
    try:
        df = pd.read_csv(path)
        print(f"Loaded '{path}' -> shape {df.shape}")
        return df
    except Exception as e:
```

```
        print(f"Failed to read {path}: {e}")
        return None


def basic_report(df, name):
    display(Markdown(f"## Report for **{name}**"))
    print("Shape:", df.shape)
    display(Markdown("**Columns & dtypes**"))
    display(pd.DataFrame(df.dtypes, columns=['dtype']))

    display(Markdown("**Head (first 5 rows)**"))
    display(df.head())

    display(Markdown("**Missing values per column**"))
    display(df.isna().sum().to_frame(name='missing'))

    display(Markdown("**Descriptive statistics**"))
    display(df.describe(include='all').transpose())


def plot_histograms(df, name, max_cols=8):
    numeric = df.select_dtypes(include=[np.number])
    if numeric.shape[1] == 0:
        print(f"No numeric columns to plot for {name}")
        return
    cols = numeric.columns.tolist()[:max_cols]
    n = len(cols)
    cols_per_row = 2
    rows = int(np.ceil(n / cols_per_row))
    fig, axes = plt.subplots(rows, cols_per_row, figsize=(6*cols_per_row,␣
 ↪3*rows))
    axes = axes.flatten()
    for i, col in enumerate(cols):
        axes[i].hist(numeric[col].dropna(), bins=30)
        axes[i].set_title(f"{col}")
        axes[i].xaxis.set_major_locator(MaxNLocator(integer=True))
    for j in range(i+1, len(axes)):
        axes[j].set_visible(False)
    fig.suptitle(f"{name} - Histograms", fontsize=14)
    plt.tight_layout(rect=[0, 0.03, 1, 0.95])
    plt.show()


def plot_boxplots(df, name, max_cols=12):
    numeric = df.select_dtypes(include=[np.number])
    if numeric.shape[1] == 0:
        print(f"No numeric columns for boxplot for {name}")
```

```
        return
    cols = numeric.columns.tolist()[:max_cols]
    fig, ax = plt.subplots(figsize=(max(6, len(cols)*1.2), 4))
    ax.boxplot([numeric[c].dropna() for c in cols], labels=cols, vert=True)
    ax.set_title(f"{name} - Boxplots")
    plt.xticks(rotation=45, ha='right')
    plt.show()


def correlation_heatmap(df, name):
    numeric = df.select_dtypes(include=[np.number])
    if numeric.shape[1] < 2:
        print(f"Not enough numeric columns for correlation heatmap for {name}")
        return
    corr = numeric.corr()
    display(Markdown(f"**Correlation matrix ({name})**"))
    display(corr)

    fig, ax = plt.subplots(figsize=(max(6, corr.shape[0]), max(4, corr.
 ↪shape[1]*0.5)))
    cax = ax.matshow(corr, vmin=-1, vmax=1)
    plt.colorbar(cax)
    ticks = range(len(corr.columns))
    ax.set_xticks(ticks)
    ax.set_yticks(ticks)
    ax.set_xticklabels(corr.columns, rotation=90)
    ax.set_yticklabels(corr.columns)
    ax.set_title(f"{name} - Correlation matrix")
    plt.tight_layout()
    plt.show()


def pca_scatter(df, name, n_components=2, label_col=None):
    numeric = df.select_dtypes(include=[np.number])
    if numeric.shape[1] == 0:
        print(f"No numeric columns for PCA for {name}")
        return
    X = numeric.fillna(numeric.mean()).values
    scaler = StandardScaler()
    Xs = scaler.fit_transform(X)
    pca = PCA(n_components=n_components)
    pc = pca.fit_transform(Xs)

    display(Markdown(f"**PCA scatter ({name}) - explained variance ratio:**␣
 ↪{pca.explained_variance_ratio_.round(3).tolist()}"))
    fig, ax = plt.subplots(figsize=(7,6))
    if label_col and label_col in df.columns:
```

```
        labels = df[label_col].astype(str).values
        unique = np.unique(labels)
        for u in unique:
            mask = labels == u
            ax.scatter(pc[mask,0], pc[mask,1], label=str(u), alpha=0.7)
        ax.legend()
    else:
        ax.scatter(pc[:,0], pc[:,1], alpha=0.7)
    ax.set_xlabel('PC1')
    ax.set_ylabel('PC2')
    ax.set_title(f"{name} - PCA scatter")
    plt.show()

# Embedding helpers

def find_embeddings_path():
    for p in EMBEDDING_PATHS:
        if os.path.exists(p):
            return p
    return None


def load_embeddings(path):
    if path.endswith('.npy'):
        emb = np.load(path)
    elif path.endswith('.csv'):
        emb = pd.read_csv(path, header=None).values
    else:
        raise ValueError('Unknown embedding format: ' + path)
    print(f"Loaded embeddings from {path} with shape {emb.shape}")
    return emb


def embeddings_report(emb, name='embeddings'):
    display(Markdown(f"## Embedding report: {name}"))
    print('Shape:', emb.shape)
    print('Mean vector norm:', np.linalg.norm(emb, axis=1).mean())
    print('Min/Max norm:', np.linalg.norm(emb, axis=1).min(), np.linalg.
 ↪norm(emb, axis=1).max())

    try:
        pca = PCA(n_components=2)
        pc = pca.fit_transform(emb)
        fig, ax = plt.subplots(figsize=(7,6))
        ax.scatter(pc[:,0], pc[:,1], alpha=0.6)
        ax.set_title('Embeddings PCA (2D)')
        plt.show()
```

```
    except Exception as e:
        print('PCA failed for embeddings:', e)

    if emb.shape[0] > 2000:
        print('Subsampling embeddings for t-SNE (2000 samples)')
        idx = np.random.choice(np.arange(emb.shape[0]), size=2000,␣
↪replace=False)
        emb_sub = emb[idx]
    else:
        emb_sub = emb
    try:
        tsne = TSNE(n_components=2, perplexity=30, n_iter=1000, init='pca',␣
↪random_state=42)
        Z = tsne.fit_transform(emb_sub)
        fig, ax = plt.subplots(figsize=(7,6))
        ax.scatter(Z[:,0], Z[:,1], alpha=0.6, s=10)
        ax.set_title('Embeddings t-SNE (2D)')
        plt.show()
    except Exception as e:
        print('t-SNE failed for embeddings:', e)
```

[18]:
```
# %%
# Cell 4 - Load datasets into notebook

dfs = {}
for key, path in DATA_FILES.items():
    df = safe_read_csv(path)
    if df is not None:
        dfs[key] = df

# %%
# Cell 5 - Run EDA interactively (this prints and shows plots inline)
for name, df in dfs.items():
    basic_report(df, name)
    plot_histograms(df, name, max_cols=8)
    plot_boxplots(df, name, max_cols=12)
    correlation_heatmap(df, name)
    label_col = None
    for candidate in ['target', 'class', 'label', 'species', 'y']:
        if candidate in df.columns:
            label_col = candidate
            break
    pca_scatter(df, name, label_col=label_col)



# End of notebook script
```

```
Loaded 'iris_openml.csv' -> shape (150, 6)
Loaded 'wine_openml.csv' -> shape (178, 15)
Loaded 'diabetes_openml.csv' -> shape (768, 10)
```

2.1 Report for iris

```
Shape: (150, 6)
```

Columns & dtypes

```
              dtype
sepallength  float64
sepalwidth   float64
petallength  float64
petalwidth   float64
class         object
target        object
```

Head (first 5 rows)

```
   sepallength  sepalwidth  petallength  petalwidth        class       target
0          5.1         3.5          1.4         0.2  Iris-setosa  Iris-setosa
1          4.9         3.0          1.4         0.2  Iris-setosa  Iris-setosa
2          4.7         3.2          1.3         0.2  Iris-setosa  Iris-setosa
3          4.6         3.1          1.5         0.2  Iris-setosa  Iris-setosa
4          5.0         3.6          1.4         0.2  Iris-setosa  Iris-setosa
```

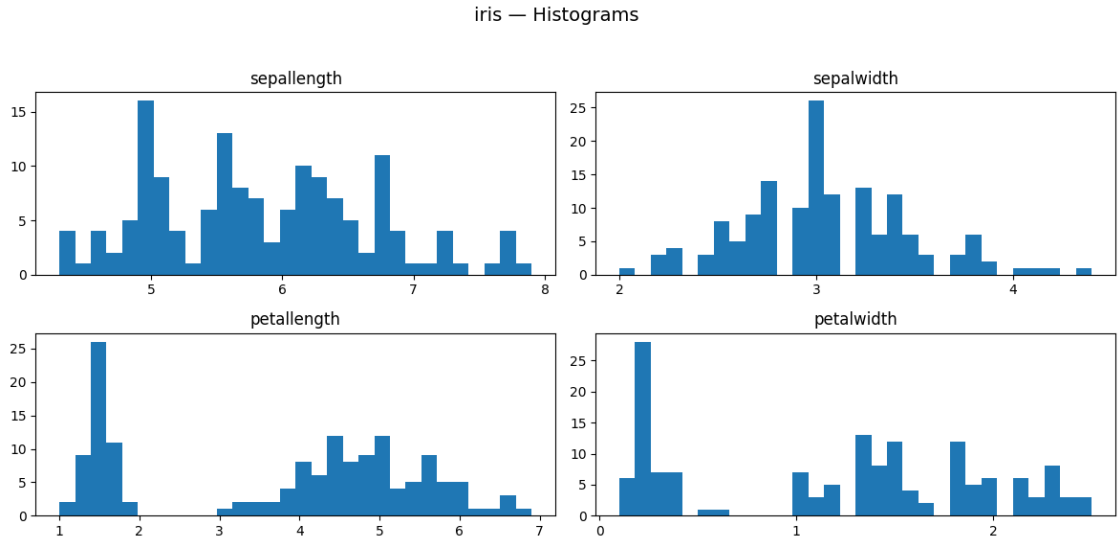
Missing values per column

```
             missing
sepallength        0
sepalwidth         0
petallength        0
petalwidth         0
class              0
target             0
```

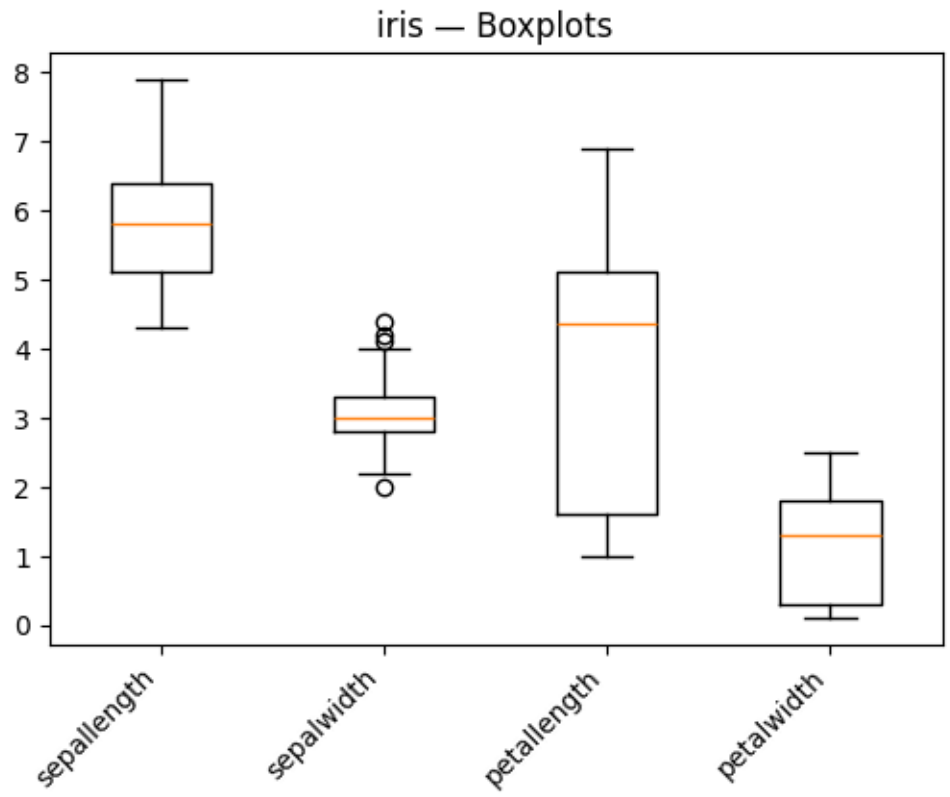
Descriptive statistics

```
             count unique          top freq      mean       std  min  25%  \
sepallength  150.0    NaN          NaN  NaN  5.843333  0.828066  4.3  5.1
sepalwidth   150.0    NaN          NaN  NaN     3.054  0.433594  2.0  2.8
petallength  150.0    NaN          NaN  NaN  3.758667   1.76442  1.0  1.6
petalwidth   150.0    NaN          NaN  NaN  1.198667  0.763161  0.1  0.3
class          150      3  Iris-setosa   50       NaN       NaN  NaN  NaN
target         150      3  Iris-setosa   50       NaN       NaN  NaN  NaN

              50%  75%  max
sepallength   5.8  6.4  7.9
sepalwidth    3.0  3.3  4.4
petallength  4.35  5.1  6.9
petalwidth    1.3  1.8  2.5
```

```
class          NaN  NaN  NaN
target         NaN  NaN  NaN
```

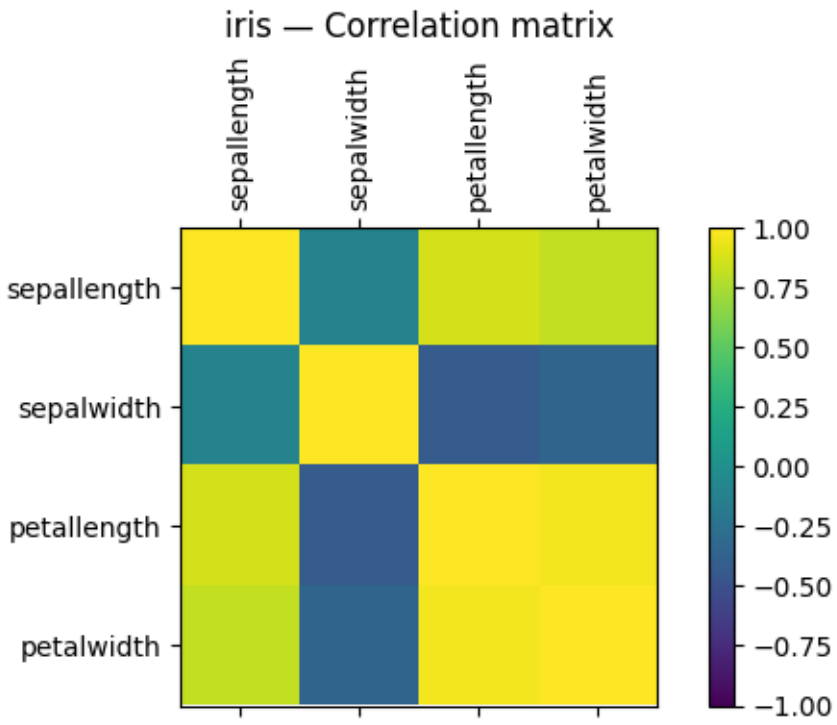



```
C:\Users\mahesh\AppData\Local\Temp\ipykernel_18308\180921521.py:89:
MatplotlibDeprecationWarning: The 'labels' parameter of boxplot() has been
renamed 'tick_labels' since Matplotlib 3.9; support for the old name will be
dropped in 3.11.
  ax.boxplot([numeric[c].dropna() for c in cols], labels=cols, vert=True)
```

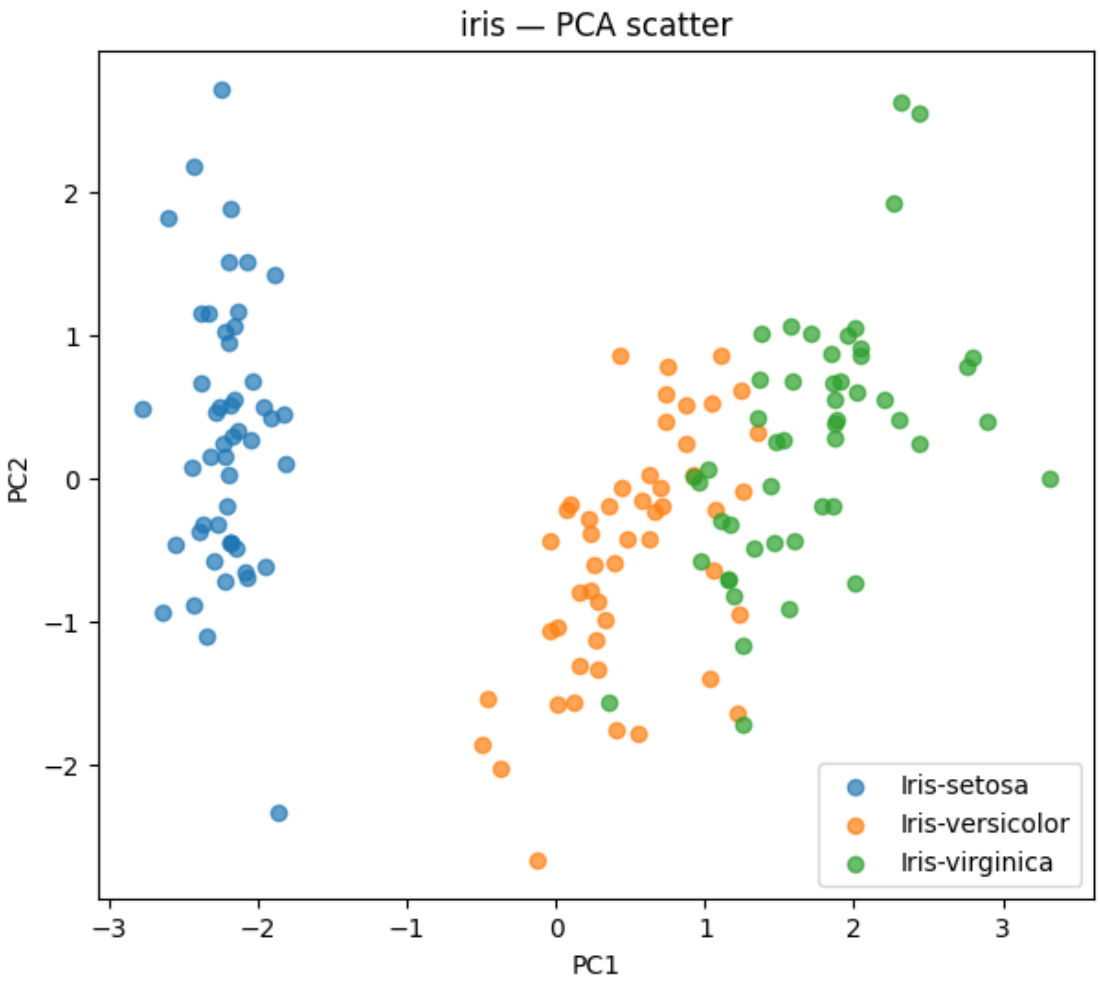


Correlation matrix (iris)

	sepallength	sepalwidth	petallength	petalwidth
sepallength	1.000000	-0.109369	0.871754	0.817954
sepalwidth	-0.109369	1.000000	-0.420516	-0.356544
petallength	0.871754	-0.420516	1.000000	0.962757
petalwidth	0.817954	-0.356544	0.962757	1.000000



PCA scatter (iris) — explained variance ratio: [0.728, 0.23]



2.2 Report for wine

Shape: (178, 15)

Columns & dtypes

	dtype
class	int64
Alcohol	float64
Malic_acid	float64
Ash	float64
Alcalinity_of_ash	float64
Magnesium	int64
Total_phenols	float64
Flavanoids	float64
Nonflavanoid_phenols	float64
Proanthocyanins	float64

```
Color_intensity                 float64
Hue                             float64
OD280%2FOD315_of_diluted_wines  float64
Proline                           int64
target                            int64
```

Head (first 5 rows)

```
   class  Alcohol  Malic_acid   Ash  Alcalinity_of_ash  Magnesium  \
0      1    14.23        1.71  2.43               15.6        127
1      1    13.20        1.78  2.14               11.2        100
2      1    13.16        2.36  2.67               18.6        101
3      1    14.37        1.95  2.50               16.8        113
4      1    13.24        2.59  2.87               21.0        118

   Total_phenols  Flavanoids  Nonflavanoid_phenols  Proanthocyanins  \
0           2.80        3.06                  0.28             2.29
1           2.65        2.76                  0.26             1.28
2           2.80        3.24                  0.30             2.81
3           3.85        3.49                  0.24             2.18
4           2.80        2.69                  0.39             1.82

   Color_intensity   Hue  OD280%2FOD315_of_diluted_wines  Proline  target
0             5.64  1.04                            3.92     1065       1
1             4.38  1.05                            3.40     1050       1
2             5.68  1.03                            3.17     1185       1
3             7.80  0.86                            3.45     1480       1
4             4.32  1.04                            2.93      735       1
```

Missing values per column

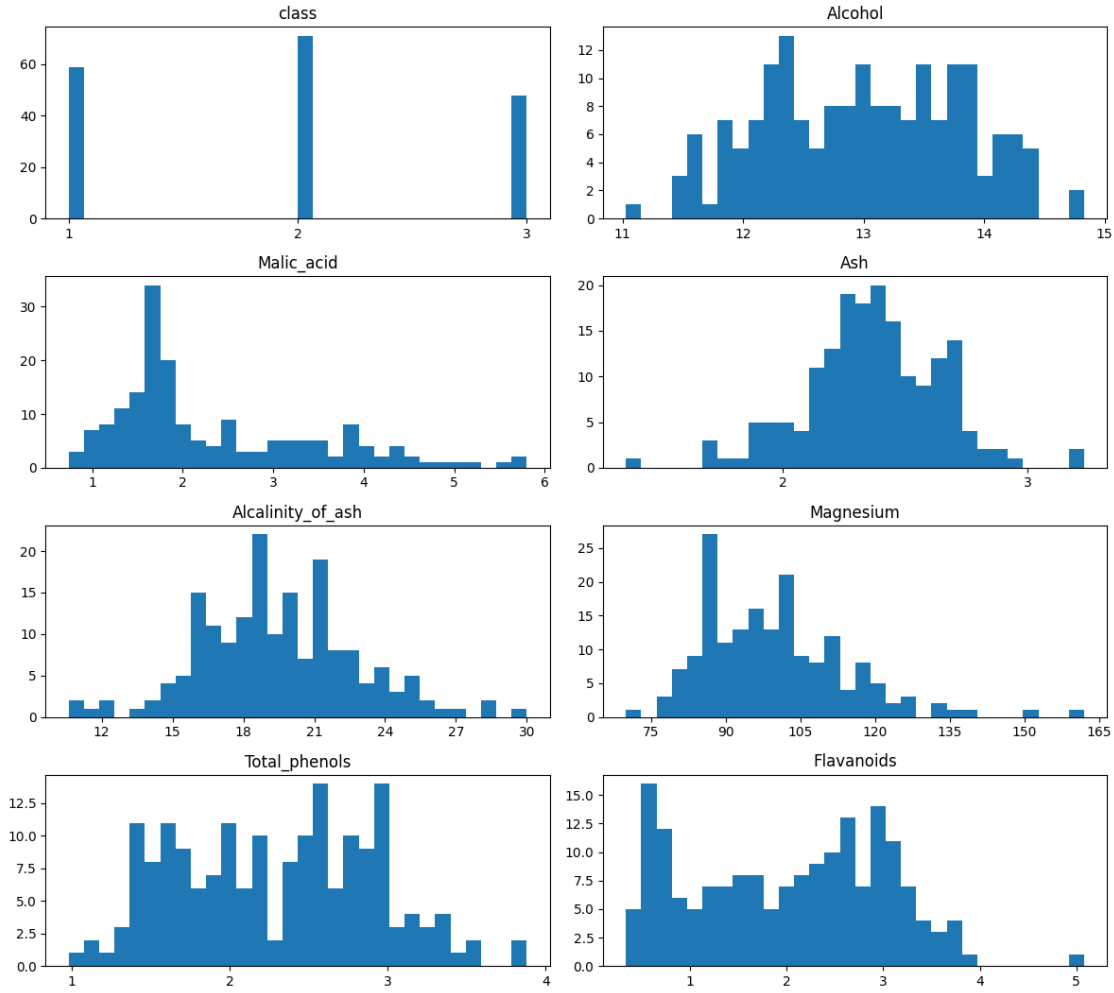
```
                                missing
class                                 0
Alcohol                               0
Malic_acid                            0
Ash                                   0
Alcalinity_of_ash                     0
Magnesium                             0
Total_phenols                         0
Flavanoids                            0
Nonflavanoid_phenols                  0
Proanthocyanins                       0
Color_intensity                       0
Hue                                   0
OD280%2FOD315_of_diluted_wines        0
Proline                               0
target                                0
```

Descriptive statistics

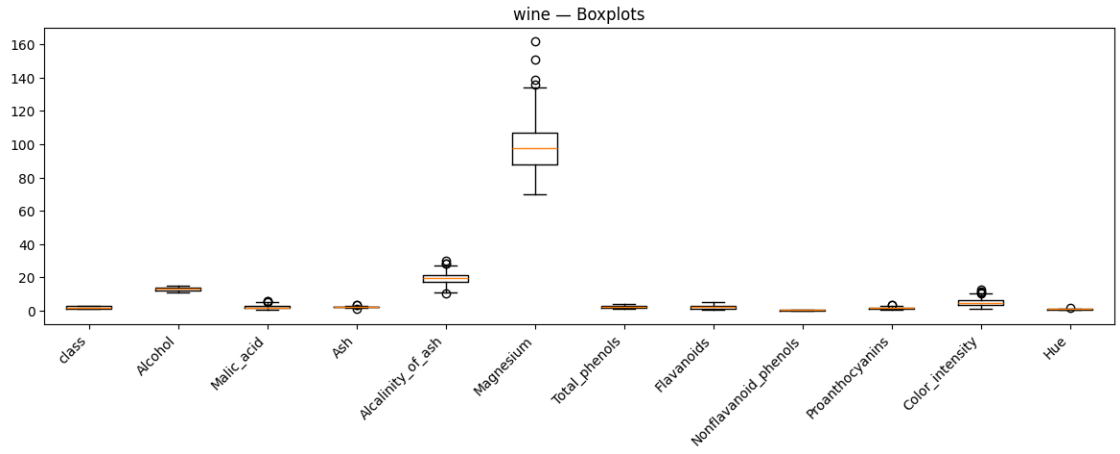
```
                                count        mean         std     min  \
```

class	178.0	1.938202	0.775035	1.00
Alcohol	178.0	13.000618	0.811827	11.03
Malic_acid	178.0	2.336348	1.117146	0.74
Ash	178.0	2.366517	0.274344	1.36
Alcalinity_of_ash	178.0	19.494944	3.339564	10.60
Magnesium	178.0	99.741573	14.282484	70.00
Total_phenols	178.0	2.295112	0.625851	0.98
Flavanoids	178.0	2.029270	0.998859	0.34
Nonflavanoid_phenols	178.0	0.361854	0.124453	0.13
Proanthocyanins	178.0	1.590899	0.572359	0.41
Color_intensity	178.0	5.058090	2.318286	1.28
Hue	178.0	0.957449	0.228572	0.48
OD280%2FOD315_of_diluted_wines	178.0	2.611685	0.709990	1.27
Proline	178.0	746.893258	314.907474	278.00
target	178.0	1.938202	0.775035	1.00

	25%	50%	75%	max
class	1.0000	2.000	3.0000	3.00
Alcohol	12.3625	13.050	13.6775	14.83
Malic_acid	1.6025	1.865	3.0825	5.80
Ash	2.2100	2.360	2.5575	3.23
Alcalinity_of_ash	17.2000	19.500	21.5000	30.00
Magnesium	88.0000	98.000	107.0000	162.00
Total_phenols	1.7425	2.355	2.8000	3.88
Flavanoids	1.2050	2.135	2.8750	5.08
Nonflavanoid_phenols	0.2700	0.340	0.4375	0.66
Proanthocyanins	1.2500	1.555	1.9500	3.58
Color_intensity	3.2200	4.690	6.2000	13.00
Hue	0.7825	0.965	1.1200	1.71
OD280%2FOD315_of_diluted_wines	1.9375	2.780	3.1700	4.00
Proline	500.5000	673.500	985.0000	1680.00
target	1.0000	2.000	3.0000	3.00




```
C:\Users\mahesh\AppData\Local\Temp\ipykernel_18308\180921521.py:89:
MatplotlibDeprecationWarning: The 'labels' parameter of boxplot() has been
renamed 'tick_labels' since Matplotlib 3.9; support for the old name will be
dropped in 3.11.
  ax.boxplot([numeric[c].dropna() for c in cols], labels=cols, vert=True)
```



Correlation matrix (wine)

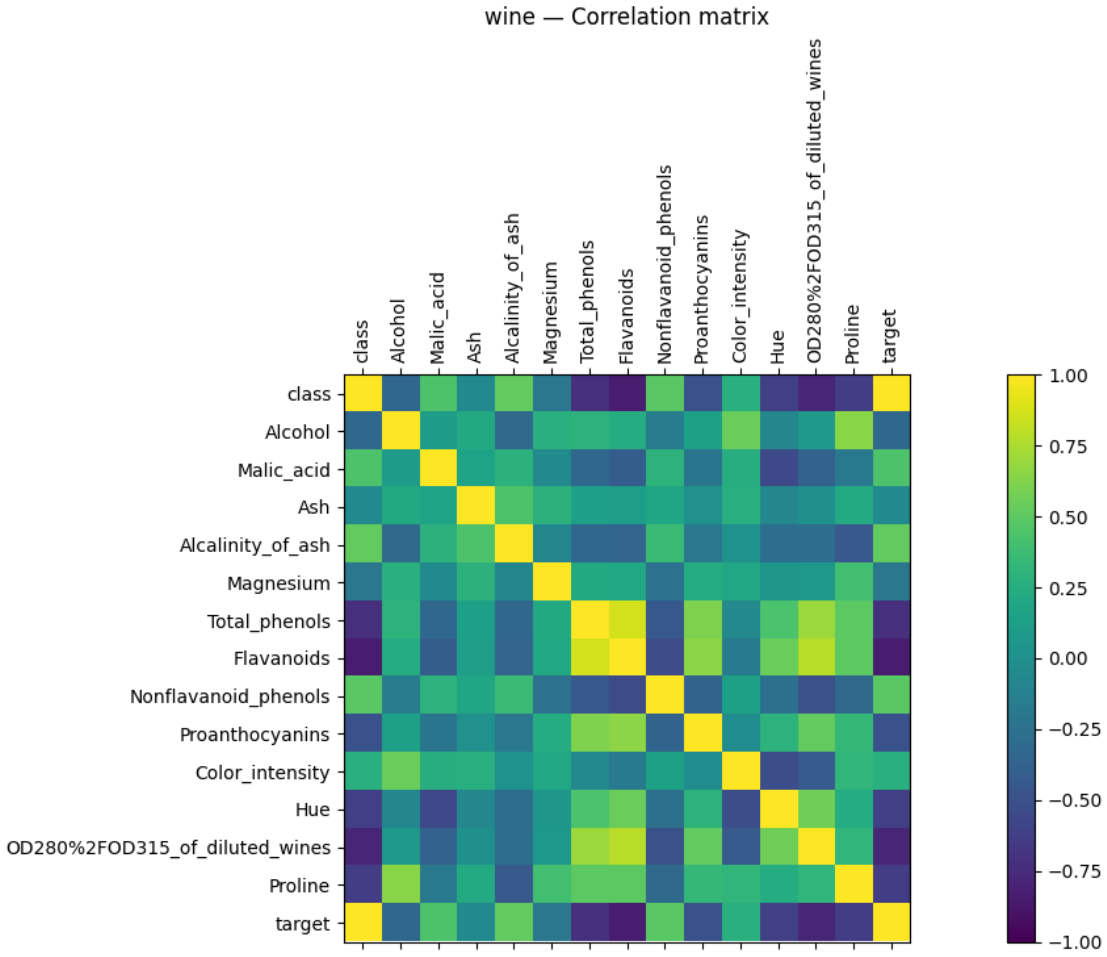
	class	Alcohol	Malic_acid	Ash	\
class	1.000000	-0.328222	0.437776	-0.049643	
Alcohol	-0.328222	1.000000	0.094397	0.211545	
Malic_acid	0.437776	0.094397	1.000000	0.164045	
Ash	-0.049643	0.211545	0.164045	1.000000	
Alcalinity_of_ash	0.517859	-0.310235	0.288500	0.443367	
Magnesium	-0.209179	0.270798	-0.054575	0.286587	
Total_phenols	-0.719163	0.289101	-0.335167	0.128980	
Flavanoids	-0.847498	0.236815	-0.411007	0.115077	
Nonflavanoid_phenols	0.489109	-0.155929	0.292977	0.186230	
Proanthocyanins	-0.499130	0.136698	-0.220746	0.009652	
Color_intensity	0.265668	0.546364	0.248985	0.258887	
Hue	-0.617369	-0.071747	-0.561296	-0.074667	
OD280%2FOD315_of_diluted_wines	-0.788230	0.072343	-0.368710	0.003911	
Proline	-0.633717	0.643720	-0.192011	0.223626	
target	1.000000	-0.328222	0.437776	-0.049643	

	Alcalinity_of_ash	Magnesium	Total_phenols	\
class	0.517859	-0.209179	-0.719163	
Alcohol	-0.310235	0.270798	0.289101	
Malic_acid	0.288500	-0.054575	-0.335167	
Ash	0.443367	0.286587	0.128980	
Alcalinity_of_ash	1.000000	-0.083333	-0.321113	
Magnesium	-0.083333	1.000000	0.214401	
Total_phenols	-0.321113	0.214401	1.000000	
Flavanoids	-0.351370	0.195784	0.864564	
Nonflavanoid_phenols	0.361922	-0.256294	-0.449935	
Proanthocyanins	-0.197327	0.236441	0.612413	
Color_intensity	0.018732	0.199950	-0.055136	
Hue	-0.273955	0.055398	0.433681	

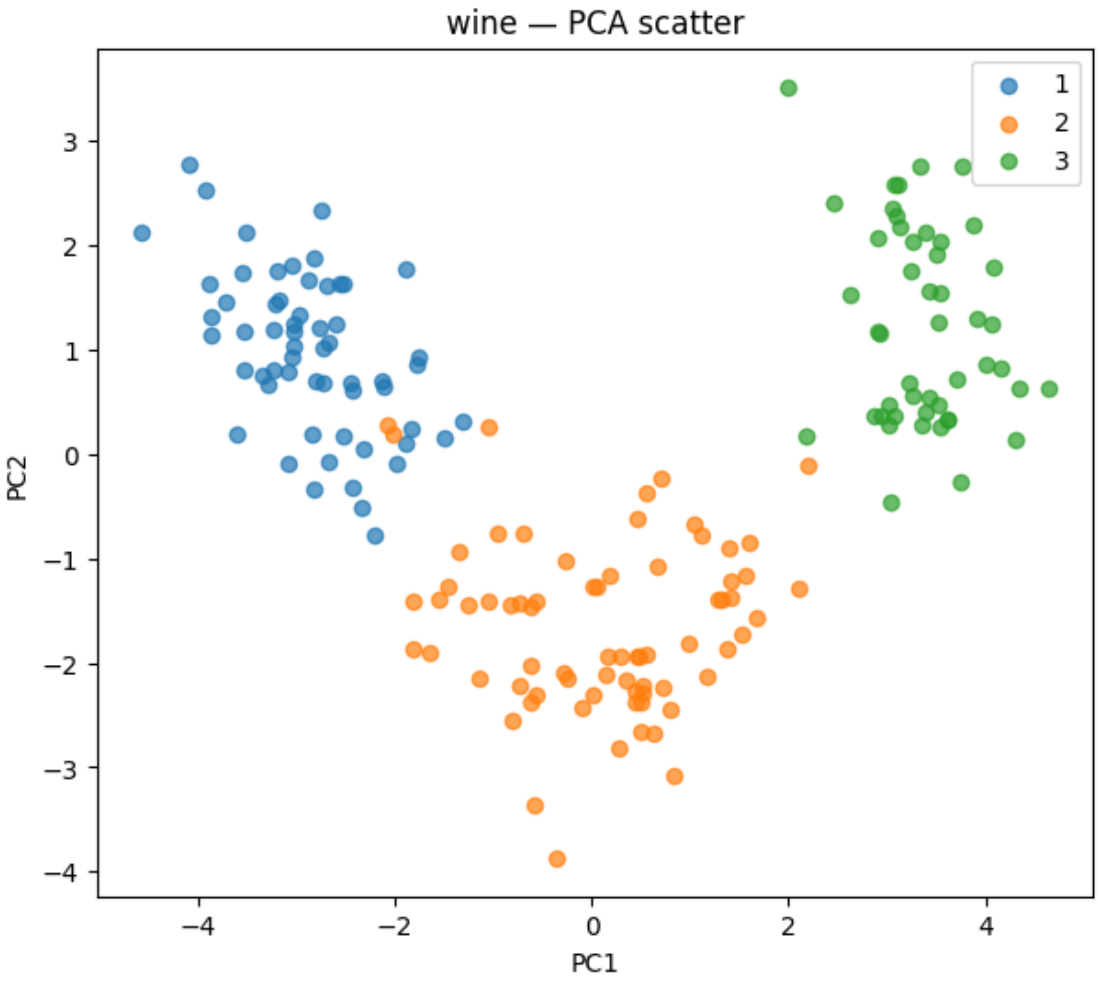
```
OD280%2FOD315_of_diluted_wines          -0.276769   0.066004       0.699949
Proline                                 -0.440597   0.393351       0.498115
target                                   0.517859  -0.209179      -0.719163

                                Flavanoids  Nonflavanoid_phenols  \
class                            -0.847498              0.489109
Alcohol                           0.236815             -0.155929
Malic_acid                       -0.411007              0.292977
Ash                               0.115077              0.186230
Alcalinity_of_ash                -0.351370              0.361922
Magnesium                         0.195784             -0.256294
Total_phenols                     0.864564             -0.449935
Flavanoids                        1.000000             -0.537900
Nonflavanoid_phenols             -0.537900              1.000000
Proanthocyanins                   0.652692             -0.365845
Color_intensity                  -0.172379              0.139057
Hue                               0.543479             -0.262640
OD280%2FOD315_of_diluted_wines    0.787194             -0.503270
Proline                           0.494193             -0.311385
target                           -0.847498              0.489109

                                Proanthocyanins  Color_intensity       Hue  \
class                                 -0.499130         0.265668 -0.617369
Alcohol                                0.136698         0.546364 -0.071747
Malic_acid                            -0.220746         0.248985 -0.561296
Ash                                    0.009652         0.258887 -0.074667
Alcalinity_of_ash                     -0.197327         0.018732 -0.273955
Magnesium                              0.236441         0.199950  0.055398
Total_phenols                          0.612413        -0.055136  0.433681
Flavanoids                             0.652692        -0.172379  0.543479
Nonflavanoid_phenols                  -0.365845         0.139057 -0.262640
Proanthocyanins                        1.000000        -0.025250  0.295544
Color_intensity                       -0.025250         1.000000 -0.521813
Hue                                    0.295544        -0.521813  1.000000
OD280%2FOD315_of_diluted_wines         0.519067        -0.428815  0.565468
Proline                                0.330417         0.316100  0.236183
target                                -0.499130         0.265668 -0.617369

                                OD280%2FOD315_of_diluted_wines   Proline  \
class                                                -0.788230 -0.633717
Alcohol                                               0.072343  0.643720
Malic_acid                                           -0.368710 -0.192011
Ash                                                   0.003911  0.223626
Alcalinity_of_ash                                    -0.276769 -0.440597
Magnesium                                             0.066004  0.393351
Total_phenols                                         0.699949  0.498115
Flavanoids                                            0.787194  0.494193
Nonflavanoid_phenols                                 -0.503270 -0.311385
```

```
Proanthocyanins                                 0.519067  0.330417
Color_intensity                                -0.428815  0.316100
Hue                                             0.565468  0.236183
OD280%2FOD315_of_diluted_wines                  1.000000  0.312761
Proline                                         0.312761  1.000000
target                                         -0.788230 -0.633717
```

```
                                target
class                         1.000000
Alcohol                      -0.328222
Malic_acid                    0.437776
Ash                          -0.049643
Alcalinity_of_ash             0.517859
Magnesium                    -0.209179
Total_phenols                -0.719163
Flavanoids                   -0.847498
Nonflavanoid_phenols          0.489109
Proanthocyanins              -0.499130
Color_intensity               0.265668
Hue                          -0.617369
OD280%2FOD315_of_diluted_wines -0.788230
Proline                      -0.633717
target                        1.000000
```



PCA scatter (wine) — explained variance ratio: [0.428, 0.166]



2.3 Report for diabetes

Shape: (768, 10)

Columns & dtypes

	dtype
preg	int64
plas	int64
pres	int64
skin	int64
insu	int64
mass	float64
pedi	float64
age	int64
class	object
target	object

Head (first 5 rows)

```
   preg  plas  pres  skin  insu  mass   pedi  age            class  \
0     6   148    72    35     0  33.6  0.627   50  tested_positive
1     1    85    66    29     0  26.6  0.351   31  tested_negative
2     8   183    64     0     0  23.3  0.672   32  tested_positive
3     1    89    66    23    94  28.1  0.167   21  tested_negative
4     0   137    40    35   168  43.1  2.288   33  tested_positive

            target
0  tested_positive
1  tested_negative
2  tested_positive
3  tested_negative
4  tested_positive
```

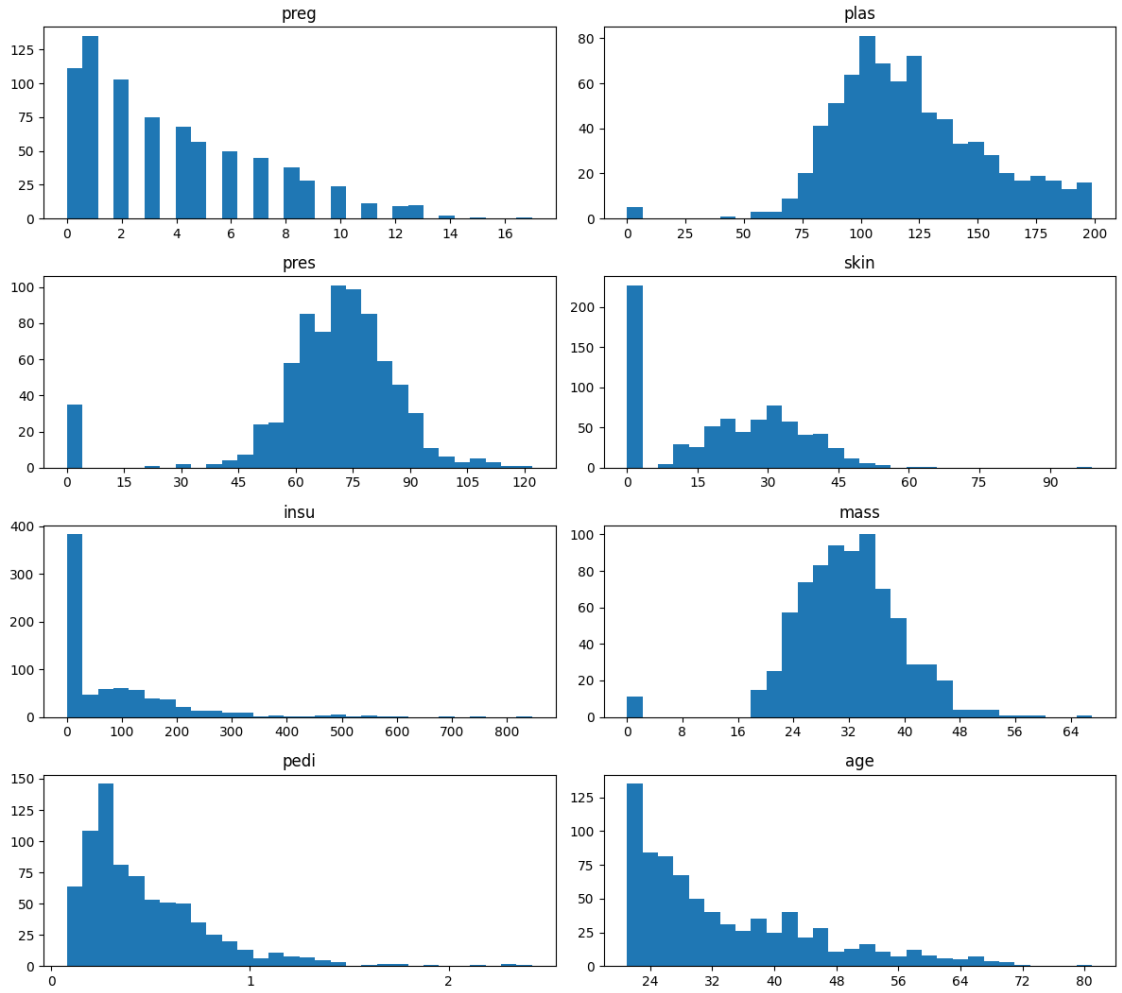
Missing values per column

```
        missing
preg          0
plas          0
pres          0
skin          0
insu          0
mass          0
pedi          0
age           0
class         0
target        0
```

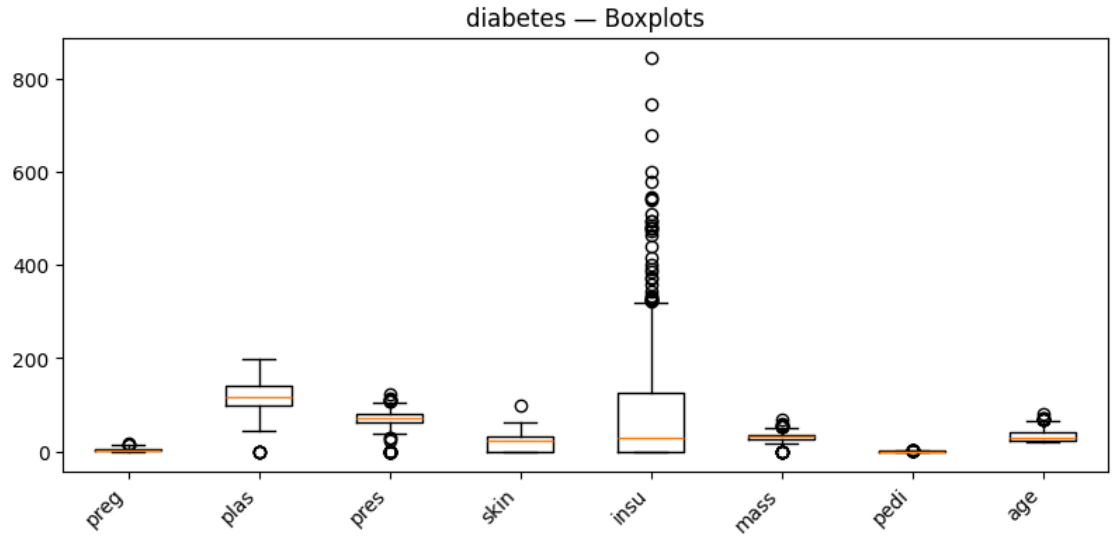
Descriptive statistics

```
        count unique              top freq        mean         std    min  \
preg    768.0    NaN              NaN  NaN    3.845052    3.369578    0.0
plas    768.0    NaN              NaN  NaN  120.894531   31.972618    0.0
pres    768.0    NaN              NaN  NaN   69.105469   19.355807    0.0
skin    768.0    NaN              NaN  NaN   20.536458   15.952218    0.0
insu    768.0    NaN              NaN  NaN   79.799479  115.244002    0.0
mass    768.0    NaN              NaN  NaN   31.992578     7.88416    0.0
pedi    768.0    NaN              NaN  NaN    0.471876    0.331329  0.078
age     768.0    NaN              NaN  NaN   33.240885   11.760232   21.0
class     768      2  tested_negative  500         NaN         NaN    NaN
target    768      2  tested_negative  500         NaN         NaN    NaN

            25%     50%      75%    max
preg        1.0     3.0      6.0   17.0
plas       99.0   117.0   140.25  199.0
pres       62.0    72.0     80.0  122.0
skin        0.0    23.0     32.0   99.0
insu        0.0    30.5   127.25  846.0
```

```
mass       27.3     32.0     36.6    67.1
pedi    0.24375   0.3725  0.62625    2.42
age        24.0     29.0     41.0    81.0
class       NaN      NaN      NaN     NaN
target      NaN      NaN      NaN     NaN
```

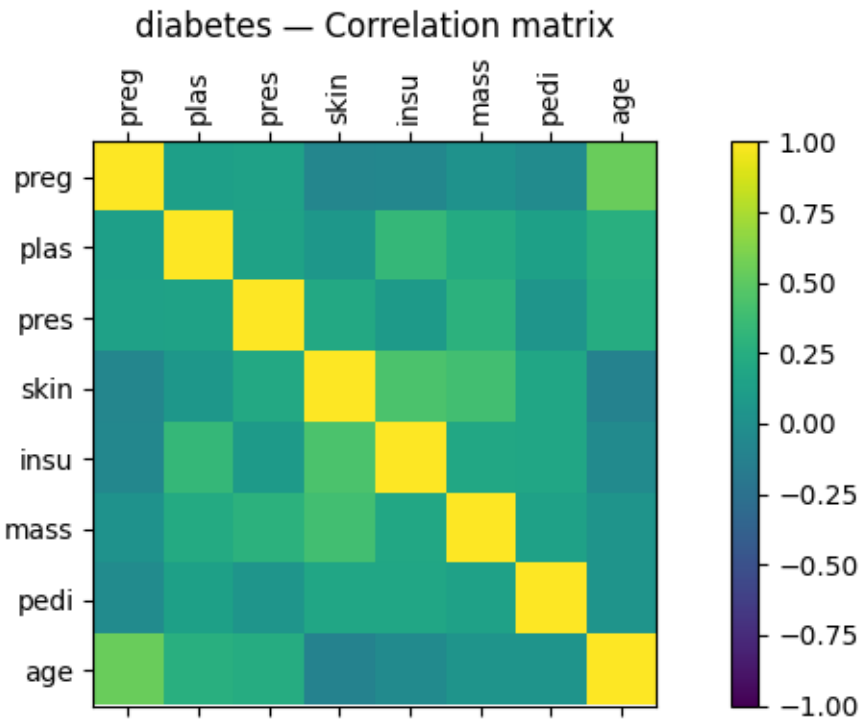



```
C:\Users\mahesh\AppData\Local\Temp\ipykernel_18308\180921521.py:89:
MatplotlibDeprecationWarning: The 'labels' parameter of boxplot() has been
renamed 'tick_labels' since Matplotlib 3.9; support for the old name will be
dropped in 3.11.
  ax.boxplot([numeric[c].dropna() for c in cols], labels=cols, vert=True)
```

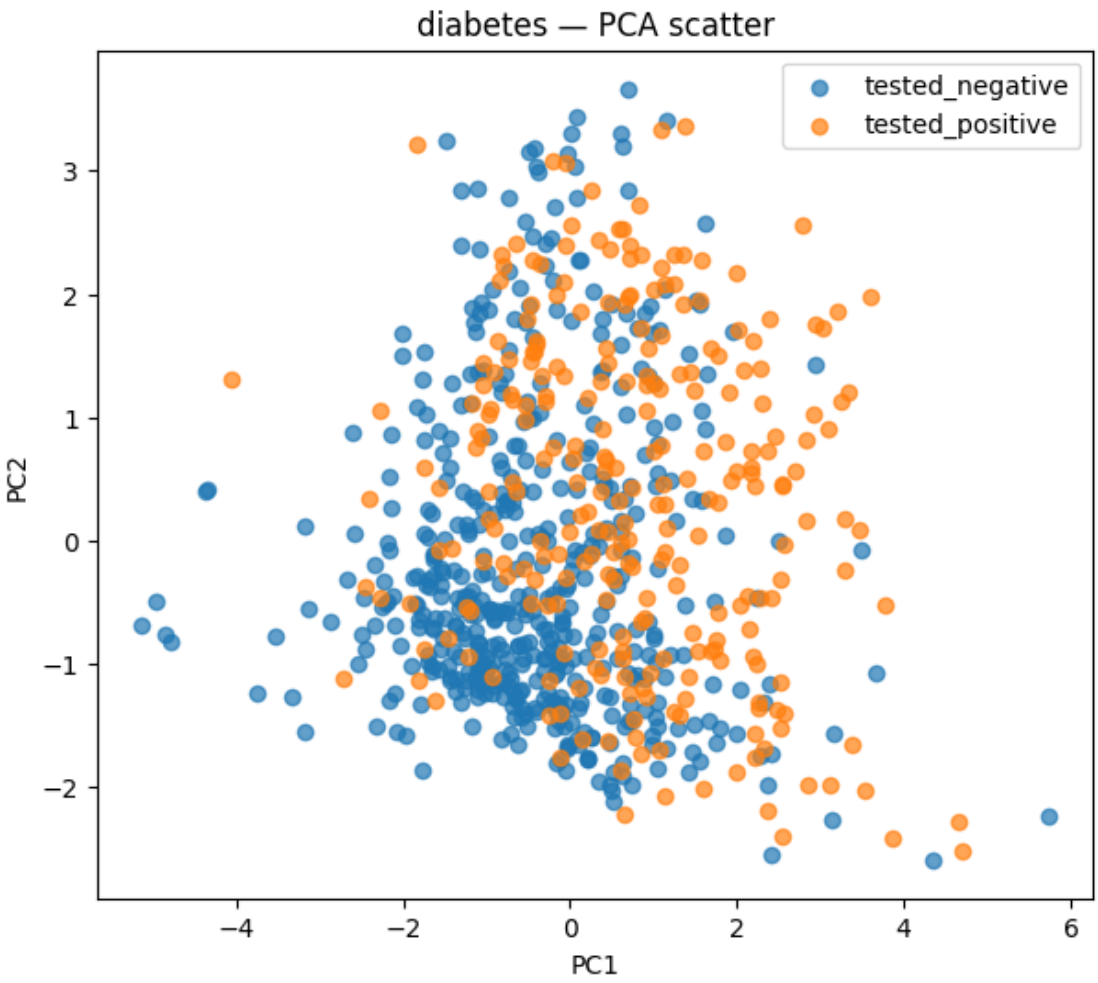


Correlation matrix (diabetes)

```
          preg      plas      pres      skin      insu      mass      pedi  \
preg  1.000000  0.129459  0.141282 -0.081672 -0.073535  0.017683 -0.033523
plas  0.129459  1.000000  0.152590  0.057328  0.331357  0.221071  0.137337
pres  0.141282  0.152590  1.000000  0.207371  0.088933  0.281805  0.041265
skin -0.081672  0.057328  0.207371  1.000000  0.436783  0.392573  0.183928
insu -0.073535  0.331357  0.088933  0.436783  1.000000  0.197859  0.185071
mass  0.017683  0.221071  0.281805  0.392573  0.197859  1.000000  0.140647
pedi -0.033523  0.137337  0.041265  0.183928  0.185071  0.140647  1.000000
age   0.544341  0.263514  0.239528 -0.113970 -0.042163  0.036242  0.033561

          age
preg  0.544341
plas  0.263514
pres  0.239528
skin -0.113970
insu -0.042163
mass  0.036242
pedi  0.033561
age   1.000000
```



PCA scatter (diabetes) — explained variance ratio: [0.262, 0.216]


diabetes — PCA scatter
tested_negative
tested_positive
PC2
PC1
3
2
1
0
−1
−2
−4
−2
0
2
4
6

[ ]: