```
      "# Importing the Libraries"
     ]
    },
    {
     "cell_type": "code",
     "metadata": {
      "id": "bAKzN8UyiScI"
     },
     "source": [
      "import numpy as np\n",
      "from sklearn import preprocessing\n",
      "from sklearn.neighbors import KNeighborsClassifier\n",
      "from sklearn.svm import SVC\n",
      "from sklearn import model_selection\n",
      "from sklearn.metrics import classification_report\n",
      "from sklearn.metrics import accuracy_score\n",
      "from pandas.plotting import scatter_matrix\n",
      "import matplotlib.pyplot as plt\n",
      "import pandas as pd\n",
      "from sklearn.model_selection import train_test_split"
     ],
     "execution_count": null,
     "outputs": []
    },
    {
     "cell_type": "markdown",
     "metadata": {
      "id": "vOthhco6spXs"
     },
     "source": [
      "# Importing the dataset"
     ]
    },
    {
     "cell_type": "code",
     "metadata": {
      "colab": {
       "base_uri": "https://localhost:8080/",
       "height": 270
      },
      "id": "iMLXz-h2oUi5",
      "outputId": "c884c84f-feb1-47b8-ca4e-d339ba295009"
     },
     "source": [
      "url=\"https://archive.ics.uci.edu/ml/machine-learning-databases/breast-cancer-
wisconsin/breast-cancer-wisconsin.data\"\n",
      "names= ['id', 'clump_thickness', 'uniform_cell_size', 'uniform_cell_shape',\n",
      "       'marginal_adhesion', 'single_epithelial_size', 'bare_nuclei',\n",
      "       'bland_chromatin', 'normal_nucleoli', 'mitoses', 'class']\n",
```

```
  "dataset=pd.read_csv(url,names=names)\n",
  "dataset.head()"
],
"execution_count": null,
"outputs": [
  {
    "output_type": "execute_result",
    "data": {
      "text/plain": [
        "       id  clump_thickness  uniform_cell_size  uniform_cell_shape  \\\n",
        "0  1000025                5                  1                   1   \n",
        "1  1002945                5                  4                   4   \n",
        "2  1015425                3                  1                   1   \n",
        "3  1016277                6                  8                   8   \n",
        "4  1017023                4                  1                   1   \n",
        "\n",
        "   marginal_adhesion  single_epithelial_size bare_nuclei  bland_chromatin  \\\n",
        "0                  1                       2           1                3   \n",
        "1                  5                       7          10                3   \n",
        "2                  1                       2           2                3   \n",
        "3                  1                       3           4                3   \n",
        "4                  3                       2           1                3   \n",
        "\n",
        "   normal_nucleoli  mitoses  class  \n",
        "0                1        1      2  \n",
        "1                2        1      2  \n",
        "2                1        1      2  \n",
        "3                7        1      2  \n",
        "4                1        1      2  "
      ],
      "text/html": [
        "\n",
        "  <div id=\"df-34551bb8-a925-4bcf-8ccd-9812cd2176f4\">\n",
        "    <div class=\"colab-df-container\">\n",
        "      <div>\n",
        "<style scoped>\n",
        "    .dataframe tbody tr th:only-of-type {\n",
        "        vertical-align: middle;\n",
        "    }\n",
        "\n",
        "    .dataframe tbody tr th {\n",
        "        vertical-align: top;\n",
        "    }\n",
        "\n",
        "    .dataframe thead th {\n",
        "        text-align: right;\n",
        "    }\n",
        "</style>\n",
        "<table border=\"1\" class=\"dataframe\">\n",
        "  <thead>\n",
```

```
"    <tr style=\"text-align: right;\">\n",
"      <th></th>\n",
"      <th>id</th>\n",
"      <th>clump_thickness</th>\n",
"      <th>uniform_cell_size</th>\n",
"      <th>uniform_cell_shape</th>\n",
"      <th>marginal_adhesion</th>\n",
"      <th>single_epithelial_size</th>\n",
"      <th>bare_nuclei</th>\n",
"      <th>bland_chromatin</th>\n",
"      <th>normal_nucleoli</th>\n",
"      <th>mitoses</th>\n",
"      <th>class</th>\n",
"    </tr>\n",
"  </thead>\n",
"  <tbody>\n",
"    <tr>\n",
"      <th>0</th>\n",
"      <td>1000025</td>\n",
"      <td>5</td>\n",
"      <td>1</td>\n",
"      <td>1</td>\n",
"      <td>1</td>\n",
"      <td>2</td>\n",
"      <td>1</td>\n",
"      <td>3</td>\n",
"      <td>1</td>\n",
"      <td>1</td>\n",
"      <td>2</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>1</th>\n",
"      <td>1002945</td>\n",
"      <td>5</td>\n",
"      <td>4</td>\n",
"      <td>4</td>\n",
"      <td>5</td>\n",
"      <td>7</td>\n",
"      <td>10</td>\n",
"      <td>3</td>\n",
"      <td>2</td>\n",
"      <td>1</td>\n",
"      <td>2</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>2</th>\n",
"      <td>1015425</td>\n",
"      <td>3</td>\n",
"      <td>1</td>\n",
"      <td>1</td>\n",
```

```
"     <td>1</td>\n",
"     <td>2</td>\n",
"     <td>2</td>\n",
"     <td>3</td>\n",
"     <td>1</td>\n",
"     <td>1</td>\n",
"     <td>2</td>\n",
"   </tr>\n",
"   <tr>\n",
"     <th>3</th>\n",
"     <td>1016277</td>\n",
"     <td>6</td>\n",
"     <td>8</td>\n",
"     <td>8</td>\n",
"     <td>1</td>\n",
"     <td>3</td>\n",
"     <td>4</td>\n",
"     <td>3</td>\n",
"     <td>7</td>\n",
"     <td>1</td>\n",
"     <td>2</td>\n",
"   </tr>\n",
"   <tr>\n",
"     <th>4</th>\n",
"     <td>1017023</td>\n",
"     <td>4</td>\n",
"     <td>1</td>\n",
"     <td>1</td>\n",
"     <td>3</td>\n",
"     <td>2</td>\n",
"     <td>1</td>\n",
"     <td>3</td>\n",
"     <td>1</td>\n",
"     <td>1</td>\n",
"     <td>2</td>\n",
"   </tr>\n",
" </tbody>\n",
"</table>\n",
"</div>\n",
"     <button class=\"colab-df-convert\" onclick=\"convertToInteractive('df-34551bb8-
a925-4bcf-8ccd-9812cd2176f4')\"\n",
"             title=\"Convert this dataframe to an interactive table.\"\n",
"             style=\"display:none;\">\n",
"       \n",
"  <svg xmlns=\"http://www.w3.org/2000/svg\" height=\"24px\"viewBox=\"0 0 24
24\"\n",
"      width=\"24px\">\n",
"   <path d=\"M0 0h24v24H0V0z\" fill=\"none\"/>\n",
"   <path d=\"M18.56 5.44l.94 2.06.94-2.06 2.06-.94-2.06-.94-.94-2.06-.94 2.06-
2.06.94zm-11 1L8.5 8.5l.94-2.06 2.06-.94-2.06-.94L8.5 2.5l-.94 2.06-2.06.94zm10 10l.94
```

```
2.06.94-2.06 2.06-.94-2.06-.94-.94-2.06-.94 2.06-2.06.94z\"/><path d=\"M17.41 7.96l-1.37-
1.37c-.4-.4-.92-.59-1.43-.59-.52 0-1.04.2-1.43.59L10.3 9.45l-7.72 7.72c-.78.78-.78 2.05 0
2.83L4 21.41c.39.39.9.59 1.41.59.51 0 1.02-.2 1.41-.59l7.78-7.78 2.81-2.81c.8-.78.8-2.07 0-
2.86zM5.41 20L4 18.59l7.72-7.72 1.47 1.35L5.41 20z\"/>\n",
          "  </svg>\n",
          "    </button>\n",
          "    \n",
          "  <style>\n",
          "    .colab-df-container {\n",
          "      display:flex;\n",
          "      flex-wrap:wrap;\n",
          "      gap: 12px;\n",
          "    }\n",
          "\n",
          "    .colab-df-convert {\n",
          "      background-color: #E8F0FE;\n",
          "      border: none;\n",
          "      border-radius: 50%;\n",
          "      cursor: pointer;\n",
          "      display: none;\n",
          "      fill: #1967D2;\n",
          "      height: 32px;\n",
          "      padding: 0 0 0 0;\n",
          "      width: 32px;\n",
          "    }\n",
          "\n",
          "    .colab-df-convert:hover {\n",
          "      background-color: #E2EBFA;\n",
          "      box-shadow: 0px 1px 2px rgba(60, 64, 67, 0.3), 0px 1px 3px 1px rgba(60, 64, 67,
0.15);\n",
          "      fill: #174EA6;\n",
          "    }\n",
          "\n",
          "    [theme=dark] .colab-df-convert {\n",
          "      background-color: #3B4455;\n",
          "      fill: #D2E3FC;\n",
          "    }\n",
          "\n",
          "    [theme=dark] .colab-df-convert:hover {\n",
          "      background-color: #434B5C;\n",
          "      box-shadow: 0px 1px 3px 1px rgba(0, 0, 0, 0.15);\n",
          "      filter: drop-shadow(0px 1px 2px rgba(0, 0, 0, 0.3));\n",
          "      fill: #FFFFFF;\n",
          "    }\n",
          "  </style>\n",
          "\n",
          "      <script>\n",
          "        const buttonEl =\n",
          "          document.querySelector('#df-34551bb8-a925-4bcf-8ccd-9812cd2176f4
button.colab-df-convert');\n",
```

```
          "        buttonEl.style.display =\n",
          "          google.colab.kernel.accessAllowed ? 'block' : 'none';\n",
          "\n",
          "        async function convertToInteractive(key) {\n",
          "          const element = document.querySelector('#df-34551bb8-a925-4bcf-8ccd-
9812cd2176f4');\n",
          "          const dataTable =\n",
          "            await google.colab.kernel.invokeFunction('convertToInteractive',\n",
          "                                                     [key], {});\n",
          "          if (!dataTable) return;\n",
          "\n",
          "          const docLinkHtml = 'Like what you see? Visit the ' +\n",
          "            '<a target=\"_blank\"
href=https://colab.research.google.com/notebooks/data_table.ipynb>data table
notebook</a>'\n",
          "            + ' to learn more about interactive tables.';\n",
          "          element.innerHTML = '';\n",
          "          dataTable['output_type'] = 'display_data';\n",
          "          await google.colab.output.renderOutput(dataTable, element);\n",
          "          const docLink = document.createElement('div');\n",
          "          docLink.innerHTML = docLinkHtml;\n",
          "          element.appendChild(docLink);\n",
          "        }\n",
          "      </script>\n",
          "    </div>\n",
          "  </div>\n",
          "  "
        ]
       },
       "metadata": {},
       "execution_count": 2
      }
     ]
    },
    {
     "cell_type": "markdown",
     "metadata": {
      "id": "ZSbi5r_Nqi5S"
     },
     "source": [
      "# cleaning the dataset"
     ]
    },
    {
     "cell_type": "code",
     "metadata": {
      "id": "QPkbL0zrqZeu",
      "colab": {
       "base_uri": "https://localhost:8080/"
      },
```

```
      "outputId": "0486b731-be98-4166-bd29-7ec75855262c"
     },
     "source": [
      "# Preprocess the data\n",
      "dataset.replace('?',-99999, inplace=True)\n",
      "print(dataset.axes)\n",
      "\n",
      "dataset.drop(['id'], 1, inplace=True)"
     ],
     "execution_count": null,
     "outputs": [
      {
       "output_type": "stream",
       "name": "stdout",
       "text": [
        "[RangeIndex(start=0, stop=699, step=1), Index(['id', 'clump_thickness',
'uniform_cell_size', 'uniform_cell_shape',\n",
        "       'marginal_adhesion', 'single_epithelial_size', 'bare_nuclei',\n",
        "       'bland_chromatin', 'normal_nucleoli', 'mitoses', 'class'],\n",
        "      dtype='object')]\n"
       ]
      },
      {
       "output_type": "stream",
       "name": "stderr",
       "text": [
        "/usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:5: FutureWarning: In a
future version of pandas all arguments of DataFrame.drop except for the argument 'labels' will
be keyword-only\n",
        "  \"\"\"\n"
       ]
      }
     ]
    },
    {
     "cell_type": "code",
     "metadata": {
      "colab": {
       "base_uri": "https://localhost:8080/"
      },
      "id": "uawtipqTHBol",
      "outputId": "077ac595-b942-4fc9-fb23-94906486e1c7"
     },
     "source": [
      "# Let explore the dataset and do a few visualizations\n",
      "print(dataset.loc[10])\n",
      "\n",
      "# Print the shape of the dataset\n",
      "print(dataset.shape)"
     ],
```

```
 "execution_count": null,
 "outputs": [
  {
   "output_type": "stream",
   "name": "stdout",
   "text": [
    "clump_thickness           1\n",
    "uniform_cell_size         1\n",
    "uniform_cell_shape        1\n",
    "marginal_adhesion         1\n",
    "single_epithelial_size    1\n",
    "bare_nuclei               1\n",
    "bland_chromatin           3\n",
    "normal_nucleoli           1\n",
    "mitoses                   1\n",
    "class                     2\n",
    "Name: 10, dtype: object\n",
    "(699, 10)\n"
   ]
  }
 ]
},
{
 "cell_type": "code",
 "metadata": {
  "colab": {
   "base_uri": "https://localhost:8080/"
  },
  "id": "JH_F5A6LHkZm",
  "outputId": "18c09058-66d3-42ee-9f6e-ae5ce95d4604"
 },
 "source": [
  "# Describe the dataset\n",
  "print(dataset.describe())"
 ],
 "execution_count": null,
 "outputs": [
  {
   "output_type": "stream",
   "name": "stdout",
   "text": [
    "       clump_thickness  uniform_cell_size  uniform_cell_shape  \\\n",
    "count       699.000000         699.000000          699.000000   \n",
    "mean          4.417740           3.134478            3.207439   \n",
    "std           2.815741           3.051459            2.971913   \n",
    "min           1.000000           1.000000            1.000000   \n",
    "25%           2.000000           1.000000            1.000000   \n",
    "50%           4.000000           1.000000            1.000000   \n",
    "75%           6.000000           5.000000            5.000000   \n",
    "max          10.000000          10.000000           10.000000   \n",
```

```
      "\n",
      "       marginal_adhesion  single_epithelial_size  bland_chromatin  \\\n",
      "count         699.000000              699.000000       699.000000   \n",
      "mean            2.806867                3.216023         3.437768   \n",
      "std             2.855379                2.214300         2.438364   \n",
      "min             1.000000                1.000000         1.000000   \n",
      "25%             1.000000                2.000000         2.000000   \n",
      "50%             1.000000                2.000000         3.000000   \n",
      "75%             4.000000                4.000000         5.000000   \n",
      "max            10.000000               10.000000        10.000000   \n",
      "\n",
      "       normal_nucleoli     mitoses       class  \n",
      "count       699.000000  699.000000  699.000000  \n",
      "mean          2.866953    1.589413    2.689557  \n",
      "std           3.053634    1.715078    0.951273  \n",
      "min           1.000000    1.000000    2.000000  \n",
      "25%           1.000000    1.000000    2.000000  \n",
      "50%           1.000000    1.000000    2.000000  \n",
      "75%           4.000000    1.000000    4.000000  \n",
      "max          10.000000   10.000000    4.000000  \n"
     ]
    }
   ]
  },
  {
   "cell_type": "code",
   "metadata": {
    "colab": {
     "base_uri": "https://localhost:8080/"
    },
    "id": "FhNDngkNP-pp",
    "outputId": "ed6d2a1f-d1d2-4799-e30f-27ae65e429b2"
   },
   "source": [
    "dataset['class'].value_counts()"
   ],
   "execution_count": null,
   "outputs": [
    {
     "output_type": "execute_result",
     "data": {
      "text/plain": [
       "2    458\n",
       "4    241\n",
       "Name: class, dtype: int64"
      ]
     },
     "metadata": {},
     "execution_count": 5
    }
```