Clean The Dataset

Project Title	Early Detection of Chronic Kidney Disease Using Machine Learning
Team Id	PNT2022TMID36376
Date	10/11/2022

```
{
  "nbformat": 4,
  "nbformat_minor": 0,
  "metadata": {
    "colab": {
      "provenance": [],
      "collapsed_sections": []
    },
    "kernelspec": {
      "name": "python3",
      "display_name": "Python 3"
    },
    "language_info": {
      "name": "python"
    }
  },
  "cells": [
    {
      "cell_type": "markdown",
      "source": [
        "**Import Packages**"
      ],
      "metadata": {
        "id": "JBMTn3poWqYH"
      }
    },
    {
      "cell_type": "code",
      "source": [
        "import pandas as pd\n",
        "import numpy as np\n",
        "import matplotlib.pyplot as plt\n",
        "import tensorflow as tf\n",
        "import tensorflow \n",
        "from tensorflow import keras\n",
        "from keras.layers import Dense"
      ],
      "metadata": {
        "id": "g1cJK-FFWwls"
      },
      "execution_count": 6,
      "outputs": []
    },
    {
      "cell_type": "markdown",
      "source": [
        "**Read dataset**"
      ],
      "metadata": {
        "id": "NNQLo9W0a16N"
      }
    },
    {
      "cell_type": "code",
      "source": [
        "data =
        pd.read_csv(\C:\Users\91801\Desktopchronickidneydisease.csv")\n",
        "print(data)"
      ],
      "metadata": {
        "colab": {
          "base_uri": "https://localhost:8080/"
        },
        "id": "FaV5jfaAa2rL",
        "outputId": "3f8335bb-02ab-40e1-a509-968b9bfa850d"
      },
      "execution_count": 43,
      "outputs": [
        {
          "output_type": "stream",
          "name": "stdout",
          "text": [
            "      id   age    bp     sg   al   su     rbc        pc
pcc  \\\n",
            "0      0  48.0  80.0  1.020  1.0  0.0     NaN    normal
```

```
notpresent   \n",
            "1      1   7.0  50.0  1.020  4.0  0.0     NaN    normal
notpresent   \n",
            "2      2  62.0  80.0  1.010  2.0  3.0  normal    normal
notpresent   \n",
            "3      3  48.0  70.0  1.005  4.0  0.0  normal  abnormal
present   \n",
            "4      4  51.0  80.0  1.010  2.0  0.0  normal    normal
notpresent   \n",
            "..   ...   ...   ...    ...  ...  ...     ...       ...
...   \n",
            "395  395  55.0  80.0  1.020  0.0  0.0  normal    normal
notpresent   \n",
            "396  396  42.0  70.0  1.025  0.0  0.0  normal    normal
notpresent   \n",
            "397  397  12.0  80.0  1.020  0.0  0.0  normal    normal
notpresent   \n",
            "398  398  17.0  60.0  1.025  0.0  0.0  normal    normal
notpresent   \n",
            "399  399  58.0  80.0  1.025  0.0  0.0  normal    normal
notpresent   \n",
            "\n",
            "             ba  ...  pcv    wc   rc  htn   dm  cad appet   pe
ane  \\\n",
            "0    notpresent  ...   44  7800  5.2  yes  yes   no  good   no
no   \n",
            "1    notpresent  ...   38  6000  NaN   no   no   no  good   no
no   \n",
            "2    notpresent  ...   31  7500  NaN   no  yes   no  poor   no
yes   \n",
            "3    notpresent  ...   32  6700  3.9  yes   no   no  poor  yes
yes   \n",
            "4    notpresent  ...   35  7300  4.6   no   no   no  good   no
no   \n",
            "..          ...  ...  ...   ...  ...  ...  ...  ...   ...  ...
...   \n",
            "395  notpresent  ...   47  6700  4.9   no   no   no  good   no
no   \n",
            "396  notpresent  ...   54  7800  6.2   no   no   no  good   no
no   \n",
            "397  notpresent  ...   49  6600  5.4   no   no   no  good   no
no   \n",
            "398  notpresent  ...   51  7200  5.9   no   no   no  good   no
no   \n",
            "399  notpresent  ...   53  6800  6.1   no   no   no  good   no
no   \n",
        "\n",
        "    classification  \n",
        "0              ckd  \n",
        "1              ckd  \n",
        "2              ckd  \n",
        "3              ckd  \n",
        "4              ckd  \n",
        "..             ...  \n",
        "395         notckd  \n",
        "396         notckd  \n",
        "397         notckd  \n",
        "398         notckd  \n",
        "399         notckd  \n",
        "\n",
        "[400 rows x 26 columns]\n"
      ]
    }
  ]
},
{
  "cell_type": "markdown",
  "source": [
    "**Understanding Data Type and Features**"
  ],
  "metadata": {
    "id": "oeh1diLw0xln"
  }
},
{
  "cell_type": "code",
  "source": [
    "print(data.info())"
  ],
  "metadata": {
    "colab": {
      "base_uri": "https://localhost:8080/"
    },
    "id": "1NPeNNkJ06P7",
    "outputId": "edf61760-725f-4220-b83b-06436b168c1f"
  },
  "execution_count": 44,
  "outputs": [
    {
      "output_type": "stream",
      "name": "stdout",
      "text": [
            "<class 'pandas.core.frame.DataFrame'>\n",
            "RangeIndex: 400 entries, 0 to 399\n",
            "Data columns (total 26 columns):\n",
            " #   Column          Non-Null Count  Dtype  \n",
            "---  ------          --------------  -----  \n",
            " 0   id              400 non-null    int64  \n",
            " 1   age             391 non-null    float64\n",
            " 2   bp              388 non-null    float64\n",
            " 3   sg              353 non-null    float64\n",
            " 4   al              354 non-null    float64\n",
            " 5   su              351 non-null    float64\n",
            " 6   rbc             248 non-null    object \n",
            " 7   pc              335 non-null    object \n",
            " 8   pcc             396 non-null    object \n",
            " 9   ba              396 non-null    object \n",
            " 10  bgr             356 non-null    float64\n",
            " 11  bu              381 non-null    float64\n",
            " 12  sc              383 non-null    float64\n",
            " 13  sod             313 non-null    float64\n",
            " 14  pot             312 non-null    float64\n",
            " 15  hemo            348 non-null    float64\n",
            " 16  pcv             330 non-null    object \n",
            " 17  wc              295 non-null    object \n",
            " 18  rc              270 non-null    object \n",
            " 19  htn             398 non-null    object \n",
            " 20  dm              398 non-null    object \n",
            " 21  cad             398 non-null    object \n",
            " 22  appet           399 non-null    object \n",
            " 23  pe              399 non-null    object \n",
            " 24  ane             399 non-null    object \n",
            " 25  classification  400 non-null    object \n",
            "dtypes: float64(11), int64(1), object(14)\n",
            "memory usage: 81.4+ KB\n",
            "None\n"
          ]
        }
      ]
    },
    {
      "cell_type": "markdown",
      "source": [
        "**Handling Missing Values**\n",
        "\n",
        "**Remove null values**"
      ],
      "metadata": {
        "id": "z_CE4RLYgMS-"
      }
    },
    {
      "cell_type": "code",
      "source": [
        "data=data.dropna(how=\"any\")\n",
        "print(data)"
      ],
      "metadata": {
        "colab": {
          "base_uri": "https://localhost:8080/"
        },
        "id": "yxiTa9MnuxXC",
        "outputId": "98f6812f-5c45-455a-9f0d-38b06b39954c"
      },
      "execution_count": 37,
      "outputs": [
        {
          "output_type": "stream",
          "name": "stdout",
          "text": [
            "      id   age    bp     sg   al   su       rbc        pc
pcc  \\\n",
            "3      3  48.0  70.0  1.005  4.0  0.0    normal  abnormal
present   \n",
            "9      9  53.0  90.0  1.020  2.0  0.0  abnormal  abnormal
present   \n",
            "11    11  63.0  70.0  1.010  3.0  0.0  abnormal  abnormal
present   \n",
            "14    14  68.0  80.0  1.010  3.0  2.0    normal  abnormal
present   \n",
            "20    20  61.0  80.0  1.015  2.0  0.0  abnormal  abnormal
notpresent   \n",
            "..   ...   ...   ...    ...  ...  ...       ...       ...
...   \n",
            "395  395  55.0  80.0  1.020  0.0  0.0    normal    normal
notpresent   \n",
            "396  396  42.0  70.0  1.025  0.0  0.0    normal    normal
notpresent   \n",
            "397  397  12.0  80.0  1.020  0.0  0.0    normal    normal
notpresent   \n",
            "398  398  17.0  60.0  1.025  0.0  0.0    normal    normal
notpresent   \n",
            "399  399  58.0  80.0  1.025  0.0  0.0    normal    normal
notpresent   \n",
            "\n",
            "             ba  ...  pcv     wc   rc  htn   dm  cad appet   pe
ane  \\\n",
            "3    notpresent  ...   32   6700  3.9  yes   no   no  poor  yes
yes   \n",
            "9    notpresent  ...   29  12100  3.7  yes  yes   no  poor   no
yes   \n",
            "11   notpresent  ...   32   4500  3.8  yes  yes   no  poor  yes
no   \n",
            "14      present  ...   16  11000  2.6  yes  yes  yes  poor  yes
no   \n",
            "20   notpresent  ...   24   9200  3.2  yes  yes  yes  poor  yes
yes   \n",
            "..          ...  ...  ...    ...  ...  ...  ...  ...   ...  ...
...   \n",
            "395  notpresent  ...   47   6700  4.9   no   no   no  good   no
no   \n",
            "396  notpresent  ...   54   7800  6.2   no   no   no  good   no
no   \n",
            "397  notpresent  ...   49   6600  5.4   no   no   no  good   no
no   \n",
            "398  notpresent  ...   51   7200  5.9   no   no   no  good   no
no   \n",
            "399  notpresent  ...   53   6800  6.1   no   no   no  good   no
no   \n",
            "\n",
            "    classification  \n",
            "3              ckd  \n",
            "9              ckd  \n",
            "11             ckd  \n",
            "14             ckd  \n",
            "20             ckd  \n",
            "..             ...  \n",
            "395         notckd  \n",
            "396         notckd  \n",
            "397         notckd  \n",
            "398         notckd  \n",
            "399         notckd  \n",
            "\n",
            "[158 rows x 26 columns]\n"
          ]
        }
      ]
    },
    {
      "cell_type": "markdown",
      "source": [
        "**Label Encoding** (String values to Numeric values)"
      ],
      "metadata": {
        "id": "eEHAMz-Uily1"
      }
    },
    {
      "cell_type": "code",
      "source": [
        "data['rbc'] = data['rbc'].map({\"abnormal\":1,\"normal\":0})\n",
        "data['pc'] = data['pc'].map({\"abnormal\":1,\"normal\":0})\n",
        "data['pcc'] = data['pcc'].map({\"present\":1,\"notpresent\":0})\n",
        "data['ba'] = data['ba'].map({\"present\":1,\"notpresent\":0})\n",
        "data['htn'] = data['htn'].map({\"yes\":1,\"no\":0})\n",
        "data['dm'] = data['dm'].map({\"yes\":1,\"no\":0})\n",
        "data['cad'] = data['cad'].map({\"yes\":1,\"no\":0})\n",
        "data['pe'] = data['pe'].map({\"yes\":1,\"no\":0})\n",
        "data['ane'] = data['ane'].map({\"yes\":1,\"no\":0})\n",
        "data['appet'] = data['appet'].map({\"poor\":1,\"good\":0})\n",
        "data['classification'] =
data['classification'].map({\"ckd\":1,\"notckd\":0})\n",
        "data['pcv'] = data['pcv'].astype('int')\n",
        "data['wc'] = data['wc'].astype('int')\n",
        "data['rc'] = data['rc'].astype('float')\n",
        "print(data)"
      ],
      "metadata": {
        "colab": {
          "base_uri": "https://localhost:8080/"
        },
        "id": "saNRgtUeu7cz",
        "outputId": "d8863f81-7919-436a-c541-953e6557ab88"
      },
      "execution_count": 38,
      "outputs": [
        {
          "output_type": "stream",
          "name": "stdout",
          "text": [
            "      id   age    bp     sg   al   su  rbc  pc  pcc  ba  ...
pcv     wc   rc  \\\n",
            "3      3  48.0  70.0  1.005  4.0  0.0    0   1    1   0  ...
32   6700  3.9   \n",
            "9      9  53.0  90.0  1.020  2.0  0.0    1   1    1   0  ...
29  12100  3.7   \n",
            "11    11  63.0  70.0  1.010  3.0  0.0    1   1    1   0  ...
32   4500  3.8   \n",
            "14    14  68.0  80.0  1.010  3.0  2.0    0   1    1   1  ...
16  11000  2.6   \n",
            "20    20  61.0  80.0  1.015  2.0  0.0    1   1    0   0  ...
24   9200  3.2   \n",
            "..   ...   ...   ...    ...  ...  ...  ...  ..  ...  ..  ...
...    ...  ...   \n",
            "395  395  55.0  80.0  1.020  0.0  0.0    0   0    0   0  ...
47   6700  4.9   \n",
            "396  396  42.0  70.0  1.025  0.0  0.0    0   0    0   0  ...
54   7800  6.2   \n",
            "397  397  12.0  80.0  1.020  0.0  0.0    0   0    0   0  ...
49   6600  5.4   \n",
            "398  398  17.0  60.0  1.025  0.0  0.0    0   0    0   0  ...
51   7200  5.9   \n",
            "399  399  58.0  80.0  1.025  0.0  0.0    0   0    0   0  ...
53   6800  6.1   \n",
            "\n",
            "     htn  dm  cad  appet  pe  ane  classification  \n",
            "3      1   0    0      1   1    1               1  \n",
            "9      1   1    0      1   0    1               1  \n",
            "11     1   1    0      1   1    0               1  \n",
            "14     1   1    1      1   1    0               1  \n",
            "20     1   1    1      1   1    1               1  \n",
            "..   ...  ..  ...    ...  ..  ...             ...  \n",
            "395    0   0    0      0   0    0               0  \n",
            "396    0   0    0      0   0    0               0  \n",
            "397    0   0    0      0   0    0               0  \n",
            "398    0   0    0      0   0    0               0  \n",
            "399    0   0    0      0   0    0               0  \n",
            "\n",
            "[158 rows x 26 columns]\n"
          ]
        }
      ]
    },
    {
      "cell_type": "markdown",
      "source": [
        "**Splitting Dependent and Independent Variable**"
      ],
      "metadata": {
        "id": "_HccszsWwGre"
      }
    },
    {
      "cell_type": "code",
      "source": [
        "X = data.iloc[:,1:25].values\n",
        "y = data.iloc[:, 25].values"
      ],
      "metadata": {
        "id": "sPD7NQ0ex9-B"
      },
      "execution_count": 39,
      "outputs": []
    },
    {
      "cell_type": "markdown",
      "source": [
        "**Split Train and Test set**"
      ],
      "metadata": {
        "id": "FXofH3PZzMXW"
      }
    },
    {
      "cell_type": "code",
      "source": [
        "from sklearn.model_selection import train_test_split\n",
        "X_train, X_test, y_train, y_test = train_test_split(X, y, test_size
= 0.30,\n",
        "random_state = 121)#101\n",
        "print(\"X train value\")\n",
        "print(X_train)\n",
        "print(\"Y train value\")\n",
        "print(y_train)"
      ],
      "metadata": {
        "colab": {
          "base_uri": "https://localhost:8080/"
        },
        "id": "bPcWZ7szzszC",
        "outputId": "0fe8a5b0-27f7-415a-8b7e-40d7c4b2f5df"
      },
      "execution_count": 41,
      "outputs": [
        {
          "output_type": "stream",
          "name": "stdout",
          "text": [
      "X train value\n",
      "[[75.    70.     1.02  ...  0.     0.     0.   ]\n",
      " [57.    60.     1.02  ...  0.     0.     0.   ]\n",
      " [66.    70.     1.025 ...  0.     0.     0.   ]\n",
      " ...\n",
      " [58.    80.     1.02  ...  0.     0.     0.   ]\n",
      " [73.    80.     1.02  ...  0.     0.     0.   ]\n",
      " [46.    60.     1.025 ...  0.     0.     0.   ]]\n",
      "Y train value\n",
      "[0 0 0 0 0 1 0 0 0 1 0 1 1 0 0 1 1 0 1 0 0 1 1 1 0 1 1 0 0 0 0 0
0 0 0 0 0\n",
      " 0 0 0 0 1 0 0 0 0 1 0 0 0 0 1 0 1 0 0 1 1 0 1 0 0 0 0 1 0 1 0 1
0 1 0 0 0\n",
      " 0 1 1 0 1 0 0 0 1 0 0 0 1 0 0 0 0 0 1 0 1 1 0 0 0 1 0 0 1 0 0 0
0 0 1 0]\n"
     ]
    }
   ]
  }
 ]
}
```