

```
In [1]:
'''
In [11]:
Project Topic: Python Machine Learning (Support Vector Machine (SVM)) analysis on "
mouse_viral_study_dataset".

Dataset:
Dataset was downloaded from Kaggle webpage. The license is shown as CC BY-SA 4.0.

Goals:
This project is totally generated for my educational and portfolio purposes to present data analysis skills in:
• Data preprocessing,
• Data analysis
• Support Vector Machine (SVM) modeling including Python, scikit-learn, and GridSearchCV
• Model evaluation
• Model visualization

Acknowledgement:
I do not own the dataset and am not redistributing it. All rights to this dataset relate to the original uploader on Kaggle.
'''

Out[11]:
'\nProject Topic: Python Machine Learning (Support Vector Machine (SVM)) analysis on "\nmouse_viral_study_dataset".\n\nDataset:\nDataset was downloaded from Kaggle webpage. The license is shown as CC BY-SA 4.0.\n\nGoals:\nThis project is totally generated for my educational and portfolio purposes to present data analysis skills in:\n\n•Data preprocessing,\n\n• Support Vector Machine (SVM) modeling including Python, scikit-learn, and GridSearchCV\n\n•Model evaluation\n\n•Model visualization\n\n\nI do not own the dataset and am not redistributing it. All rights to this dataset relate to the original uploader on Kaggle.\n\n'

In [13]:
import pandas as pd
df = pd.read_csv("mouse_viral_study.csv")
print(df.head)

In [15]:
import seaborn as sns

# Visualize the pattern between x= "Med_1_mL" and y="Med_2_mL
sns.scatterplot(data = df, x= "Med_1_mL", y="Med_2_mL", style="Virus Present", sizer=(15, 1000), alpha=0.7, hue = "Virus Present")

# Show plot
import matplotlib.pyplot as plt

plt.show()

In [6]:
# Make and then plot a linear hyperplane

import numpy as np
x = np.linspace(0, 15, 80)
m = -1.5 # Slope
c = 15 # Intercept
y = m*x + c # This is a hyperplane (a line)
plt.plot(x, y, "blue")
plt.show()

In [19]:
# Train and then visualize a Support Vector Classifier (SVC) with the ("mouse_viral_study.csv") file

# Load SVC from Support Vector Machines (SVM)
from sklearn.svm import SVC
#help(SVC)

# Look at two features (Med_1_mL and Med_2_mL)
X = df.drop("Virus Present", axis = 1)
y = df["Virus Present"]

# Initialize the model kernel = "linear" and a regularization parameter (c) = 900
model = SVC(kernel = "linear", C = 900 )

# Fit model
model.fit(X, y)

# Load plot_svm_boundary(model,X,y) and then visualize the SVM boundary.
from svm_margin_plot import plot_svm_boundary
plot_svm_boundary(model, X, y)

C:\Users\gobin\anaconda3\lib\site-packages\sklearn\base.py:493: UserWarning: X does not have valid feature names, but SVC was fitted with feature names
warnings.warn(

In [11]:
# Redo the plot with a different parameter, c value (e. g. 0.03)

model = SVC(kernel = "linear", C = 0.03)
model.fit(X, y)
plot_svm_boundary(model, X, y)

C:\Users\gobin\anaconda3\lib\site-packages\sklearn\base.py:493: UserWarning: X does not have valid feature names, but SVC was fitted with feature names
warnings.warn(

In [13]:
# Redo the plot with a different parameter, c value (e. g. 0.001)

model = SVC(kernel = "linear", C = 0.001)
model.fit(X, y)
plot_svm_boundary(model, X, y) # more data points within the margins.

C:\Users\gobin\anaconda3\lib\site-packages\sklearn\base.py:493: UserWarning: X does not have valid feature names, but SVC was fitted with feature names
warnings.warn(

In [15]:
# Use the Radial basis function (rbf) as a kernel

model = SVC(kernel = "rbf", C = 2)
model.fit(X, y)
plot_svm_boundary(model, X, y)

C:\Users\gobin\anaconda3\lib\site-packages\sklearn\base.py:493: UserWarning: X does not have valid feature names, but SVC was fitted with feature names
warnings.warn(

In [17]:
# Plot with kernel = "rbf", and gamma = "scale")

model = SVC(kernel = "rbf", C = 2, gamma = "scale")
model.fit(X, y)
plot_svm_boundary(model, X, y)

C:\Users\gobin\anaconda3\lib\site-packages\sklearn\base.py:493: UserWarning: X does not have valid feature names, but SVC was fitted with feature names
warnings.warn(

In [19]:
# Look at a different c value

model = SVC(kernel = "rbf", C = 1, gamma = "auto")
model.fit(X, y)
plot_svm_boundary(model, X, y) # Creating a noise

C:\Users\gobin\anaconda3\lib\site-packages\sklearn\base.py:493: UserWarning: X does not have valid feature names, but SVC was fitted with feature names
warnings.warn(

In [21]:
# Look at different gamma value.

model = SVC(kernel = "rbf", C = 1.5, gamma = 0.8)
model.fit(X, y)
plot_svm_boundary(model, X, y) # Making a lot of noise.

C:\Users\gobin\anaconda3\lib\site-packages\sklearn\base.py:493: UserWarning: X does not have valid feature names, but SVC was fitted with feature names
warnings.warn(

In [23]:
# Seen overfitting data at higher gamma value

model = SVC(kernel = "rbf", C = 1.5, gamma = 2.2)
model.fit(X, y)
plot_svm_boundary(model, X, y)

C:\Users\gobin\anaconda3\lib\site-packages\sklearn\base.py:493: UserWarning: X does not have valid feature names, but SVC was fitted with feature names
warnings.warn(

In [25]:
# Apply kernel = "sigmoid".

model = SVC(kernel = "sigmoid")
model.fit(X, y)
plot_svm_boundary(model, X, y)

C:\Users\gobin\anaconda3\lib\site-packages\sklearn\base.py:493: UserWarning: X does not have valid feature names, but SVC was fitted with feature names
warnings.warn(

In [27]:
# Apply polynomial kernel with degree = 1 (a linear).

model = SVC(kernel = "poly", C = 1, degree = 1)
model.fit(X, y)
plot_svm_boundary(model, X, y)

C:\Users\gobin\anaconda3\lib\site-packages\sklearn\base.py:493: UserWarning: X does not have valid feature names, but SVC was fitted with feature names
warnings.warn(

In [29]:
# polynomial kernel with degree = 3

model = SVC(kernel = "poly", C = 1.5, degree = 3)
model.fit(X, y)
plot_svm_boundary(model, X, y)

C:\Users\gobin\anaconda3\lib\site-packages\sklearn\base.py:493: UserWarning: X does not have valid feature names, but SVC was fitted with feature names
warnings.warn(

In [31]:
# polynomial with degree = 0.02 and C = 0.05

model = SVC(kernel = "poly", C = 0.02, degree = 3)
model.fit(X, y)
plot_svm_boundary(model, X, y)

C:\Users\gobin\anaconda3\lib\site-packages\sklearn\base.py:493: UserWarning: X does not have valid feature names, but SVC was fitted with feature names
warnings.warn(

In [33]:
# polynomial kernel with degree = 4 and C = 0.05.

model = SVC(kernel = "poly", C = 0.03, degree = 4)
model.fit(X, y)
plot_svm_boundary(model, X, y)

C:\Users\gobin\anaconda3\lib\site-packages\sklearn\base.py:493: UserWarning: X does not have valid feature names, but SVC was fitted with feature names
warnings.warn(

In [38]:
# Load GridSearchCV
from sklearn.model_selection import GridSearchCV

# Make a Support Vector Classifier (SVM) model
svm = SVC()

# Define parameter grid
param_grid = {"C": [0.03, 0.6, 1.2], "kernel": ["linear", "rbf"]}

# Initialize GridSearchCV
grid = GridSearchCV(svm, param_grid)

# Fit the grid
grid.fit(X, y)

# Receive the best parameters
grid.best_params_

print(grid.best_params_)
```


