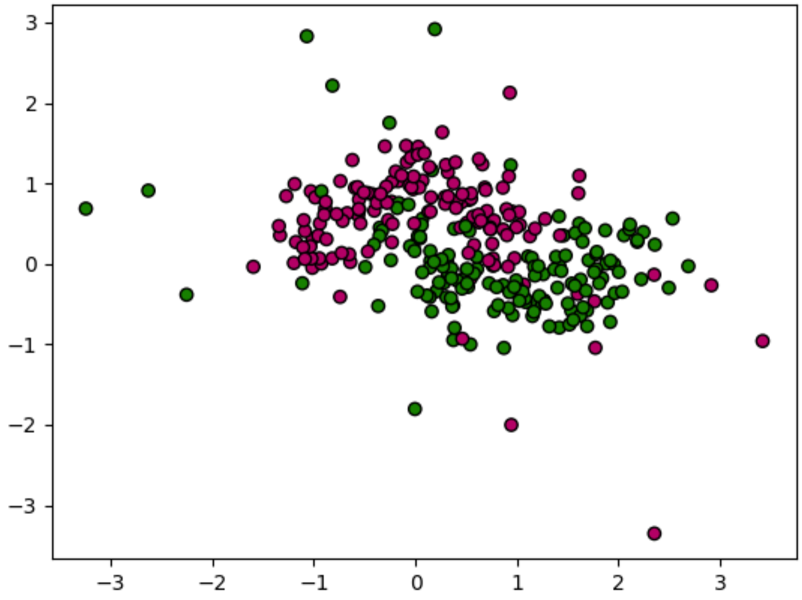
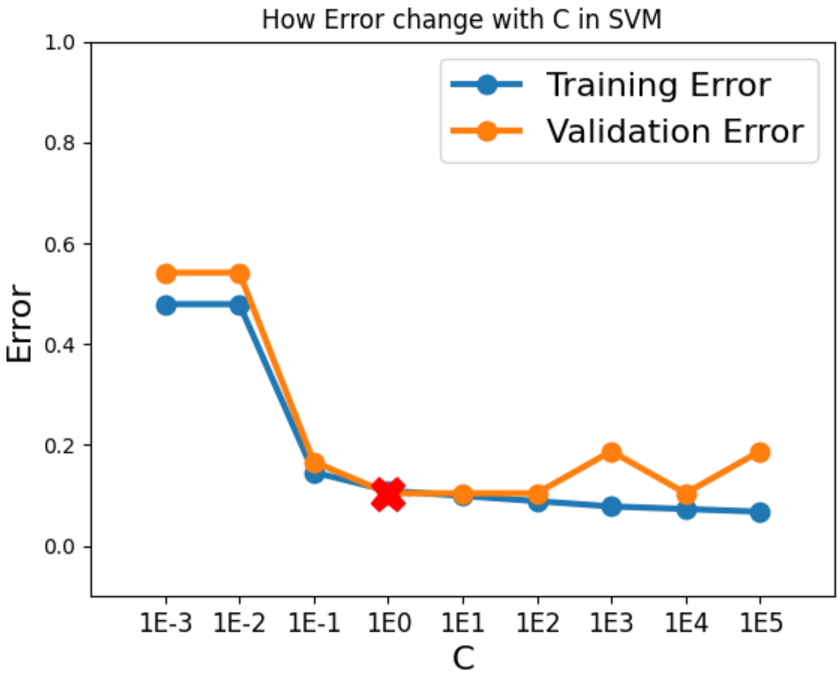
Assignment 4



1. Support Vector Machines with Synthetic Data

a. The effect of the regularization parameter C

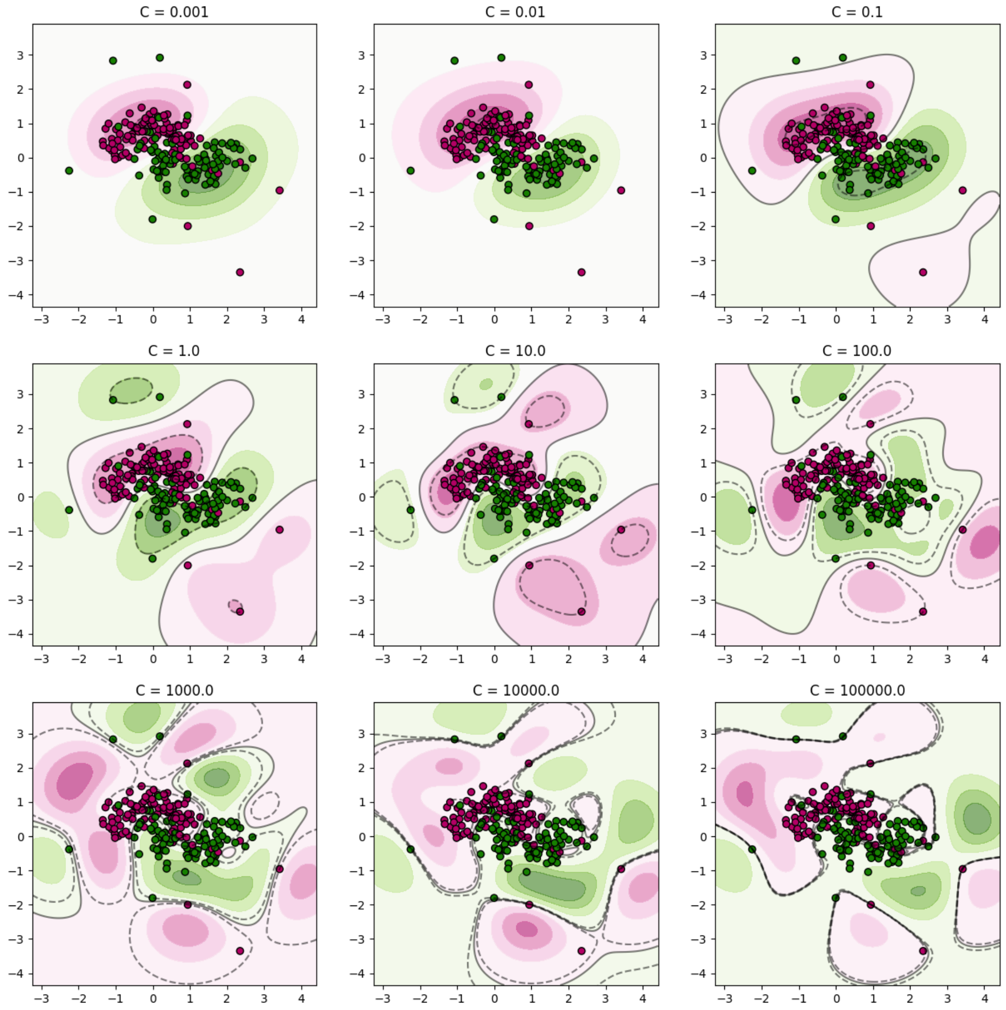
Plot:



Discussion:

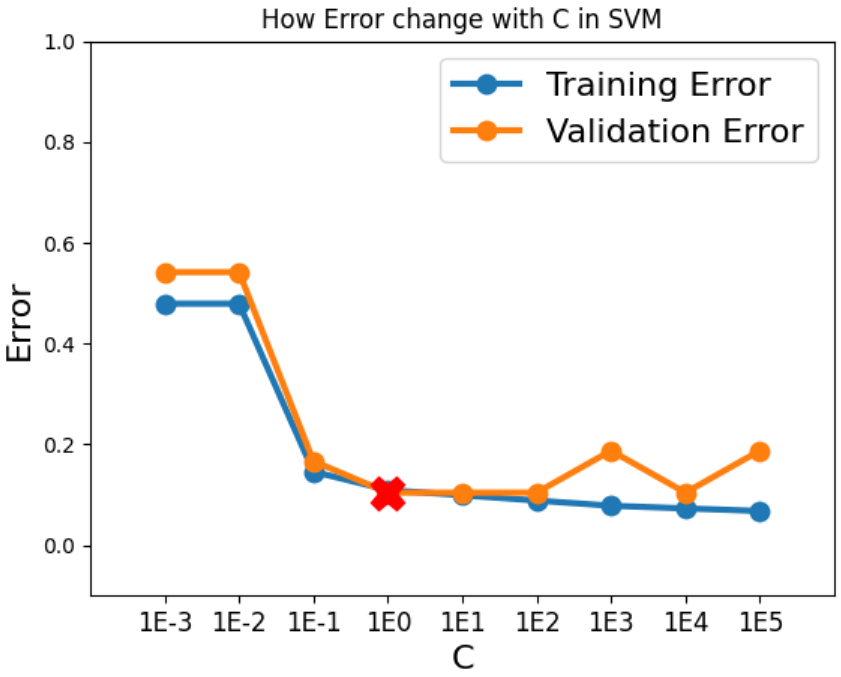
Training Error: Training Error monotonically decreases while C increases. It decreases quickly at first and slowly in the end.

Validation Error: Validation Error decreases when C <= 1 and generally increases afterwards (overfitting).



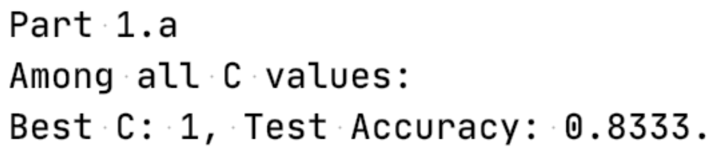
C is a trade-off between training error and flatness. Some people called C as Cross-validation parameter. While C increases, the model is softer, which means more slack are allowed and whole model is more precise and less misclassifying. Vice versa.

Final Model Selection:



To have the least Validation Error, C_{best} = 1.

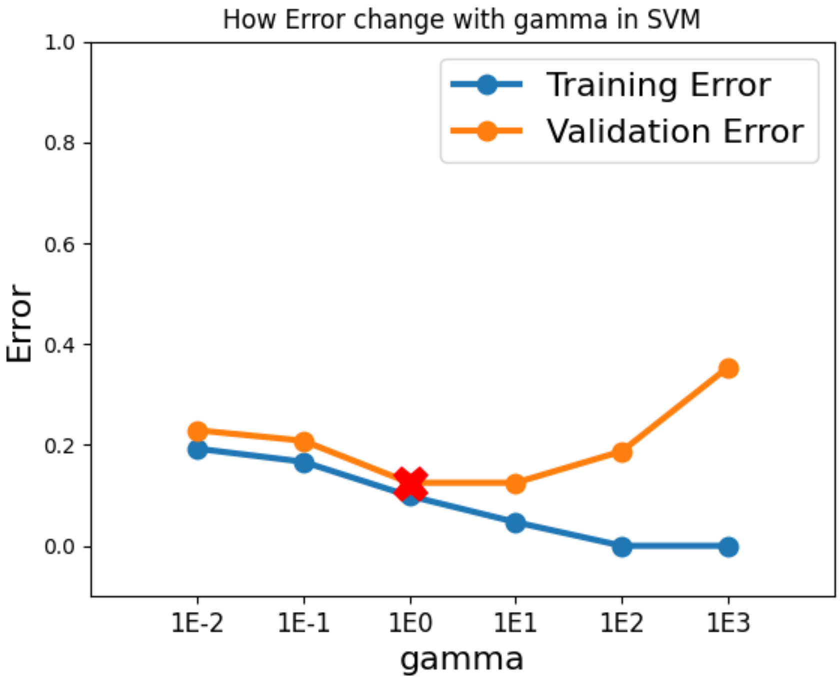
Console:



```
Part 1.a
Among all C values:
Best C: 1, Test Accuracy: 0.8333.
```

b. The effect of RBF kernel parameter γ

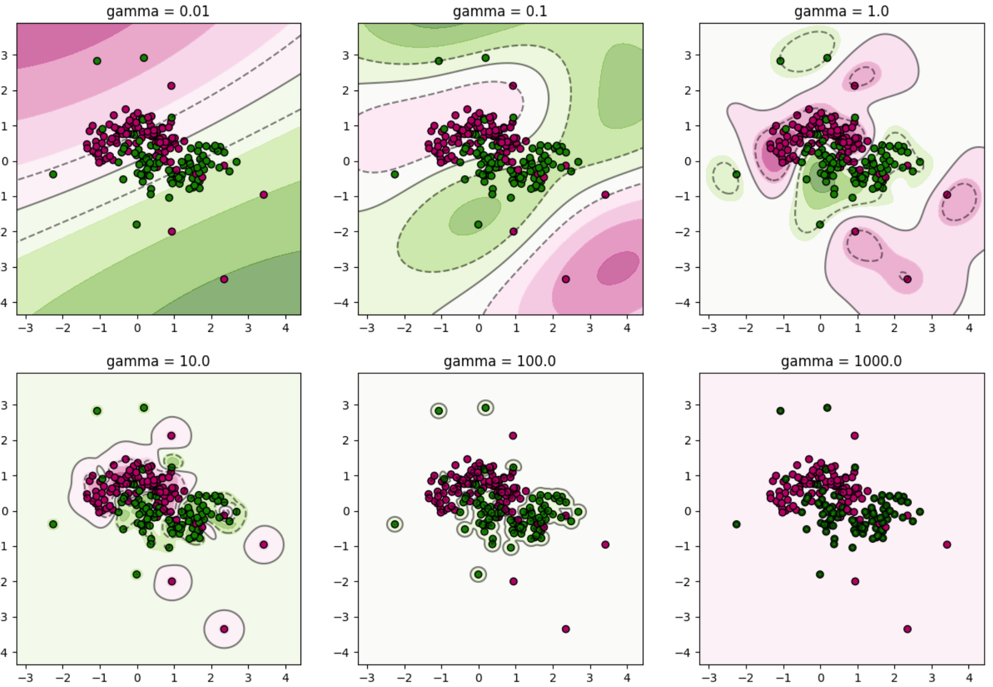
Plot:



Discussion:

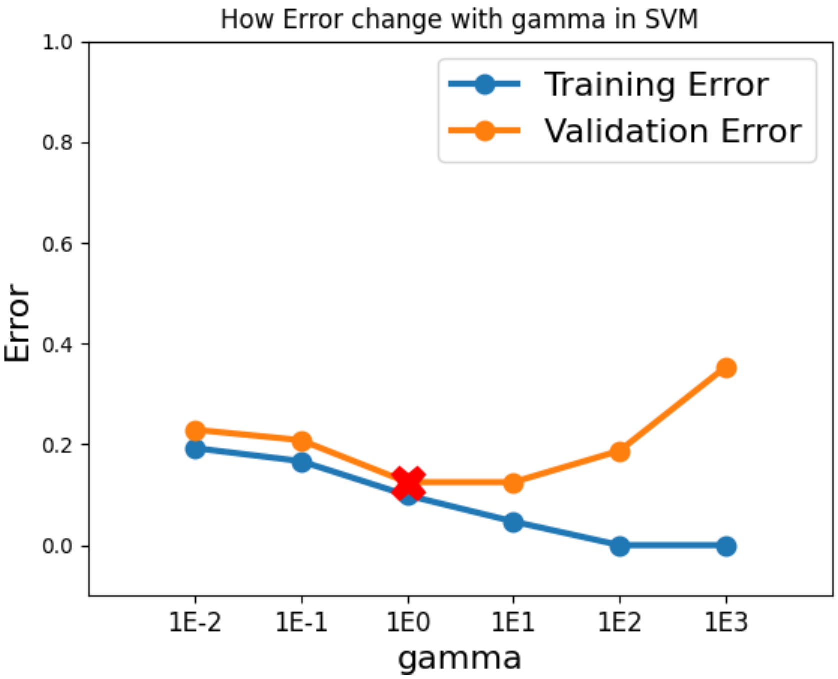
Training Error: Training Error monotonically decreases while $\boldsymbol{\gamma}$ increases. It decreases quickly at first and slowly in the end.

Validation Error: Validation Error decreases when $\boldsymbol{\gamma}$ <= 1 and generally increases afterwards (overfitting).

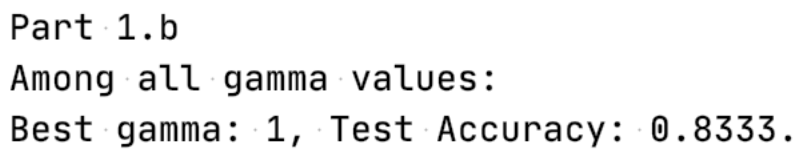


γ defines how far the influence of a single training example reaches. For a big **γ**, it will generate a sharp heap which will locate most of its contribution near the center. Hence, less constrain will cause the model loss the sense of the overall shape of data. When **γ** is large enough, the model's accuracy is close to 1 but useless for classification.

Final Model Selection:



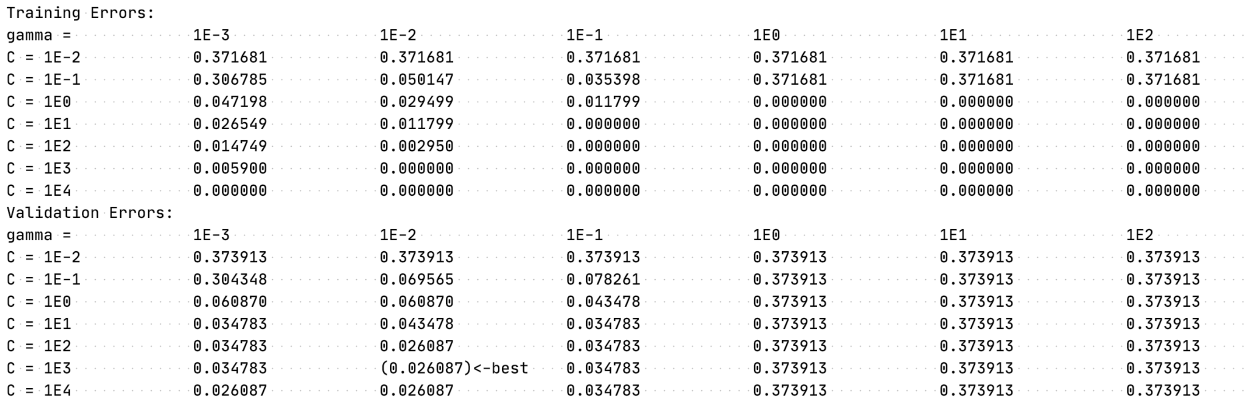
To have the least Validation Error, γ_{best} = 1.



```
Part 1.b
Among all gamma values:
Best gamma: 1, Test Accuracy: 0.8333.
```

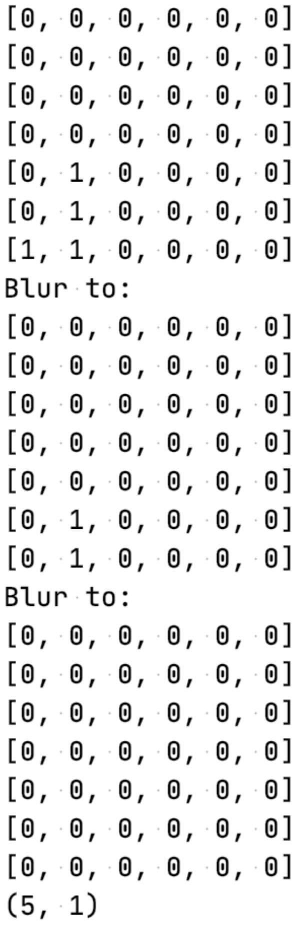
2. Breast Cancer Diagnosis with Support Vector Machines

Print Errors:



```
Training Errors:
gamma =         1E-3            1E-2            1E-1            1E0             1E1             1E2
C = 1E-2        0.371681        0.371681        0.371681        0.371681        0.371681        0.371681
C = 1E-1        0.306785        0.050147        0.035398        0.371681        0.371681        0.371681
C = 1E0         0.047198        0.029499        0.011799        0.000000        0.000000        0.000000
C = 1E1         0.026549        0.011799        0.000000        0.000000        0.000000        0.000000
C = 1E2         0.014749        0.002950        0.000000        0.000000        0.000000        0.000000
C = 1E3         0.005900        0.000000        0.000000        0.000000        0.000000        0.000000
C = 1E4         0.000000        0.000000        0.000000        0.000000        0.000000        0.000000
Validation Errors:
gamma =         1E-3            1E-2            1E-1            1E0             1E1             1E2
C = 1E-2        0.373913        0.373913        0.373913        0.373913        0.373913        0.373913
C = 1E-1        0.304348        0.069565        0.078261        0.373913        0.373913        0.373913
C = 1E0         0.060870        0.060870        0.043478        0.373913        0.373913        0.373913
C = 1E1         0.034783        0.043478        0.034783        0.373913        0.373913        0.373913
C = 1E2         0.034783        0.026087        0.034783        0.373913        0.373913        0.373913
C = 1E3         0.034783        (0.026087)<-best 0.034783       0.373913        0.373913        0.373913
C = 1E4         0.026087        0.026087        0.034783        0.373913        0.373913        0.373913
```

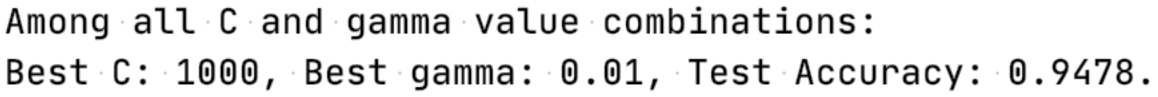
I use median to blur the matrix and find the final “best” (C, γ) pair.



```
[0, 0, 0, 0, 0, 0]
[0, 0, 0, 0, 0, 0]
[0, 0, 0, 0, 0, 0]
[0, 0, 0, 0, 0, 0]
[0, 1, 0, 0, 0, 0]
[0, 1, 0, 0, 0, 0]
[1, 1, 0, 0, 0, 0]
Blur to:
[0, 0, 0, 0, 0, 0]
[0, 0, 0, 0, 0, 0]
[0, 0, 0, 0, 0, 0]
[0, 0, 0, 0, 0, 0]
[0, 0, 0, 0, 0, 0]
[0, 1, 0, 0, 0, 0]
[0, 1, 0, 0, 0, 0]
Blur to:
[0, 0, 0, 0, 0, 0]
[0, 0, 0, 0, 0, 0]
[0, 0, 0, 0, 0, 0]
[0, 0, 0, 0, 0, 0]
[0, 0, 0, 0, 0, 0]
[0, 0, 0, 0, 0, 0]
[0, 0, 0, 0, 0, 0]
(5, 1)
```

Final Model Selection:

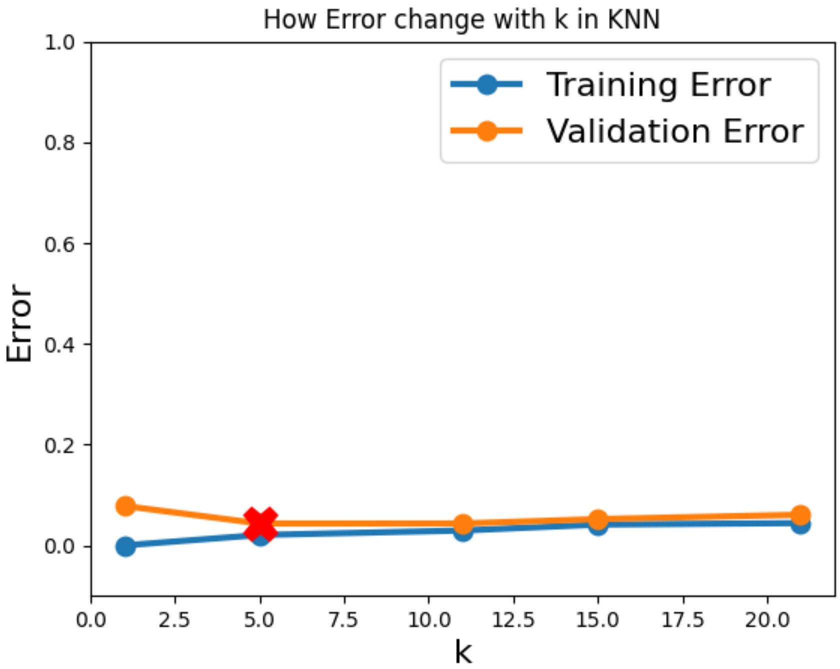
To have the least Validation Error, C_{best} = 1000 and $\boldsymbol{\gamma}_{best}$ = 0.01.



```
Among all C and gamma value combinations:
Best C: 1000, Best gamma: 0.01, Test Accuracy: 0.9478.
```

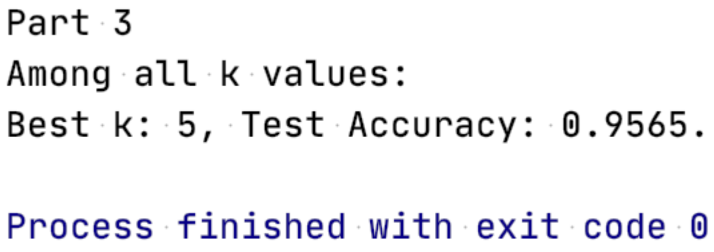
3. Breast Cancer Diagnosis with k-Nearest Neighbors

Plot:



Final Model Selection:

To have the least Validation Error, k_{best} = 5.



```
Part 3
Among all k values:
Best k: 5, Test Accuracy: 0.9565.

Process finished with exit code 0
```

Discussion:

Depending on the result which I got, I will prefer to use kNN. For the test accuracy of kNN in k_{best} is larger than that of SVM in (C_{best}, γ_{best}). But all these two is good for Breast Cancer Diagnosis: Test Accuracy(SVMs(C_{best}=1000, $\gamma_{best}=0.01$)) $= 0.9478 < 0.9565 =$ Test Accuracy(kNN(k_{best}=5))