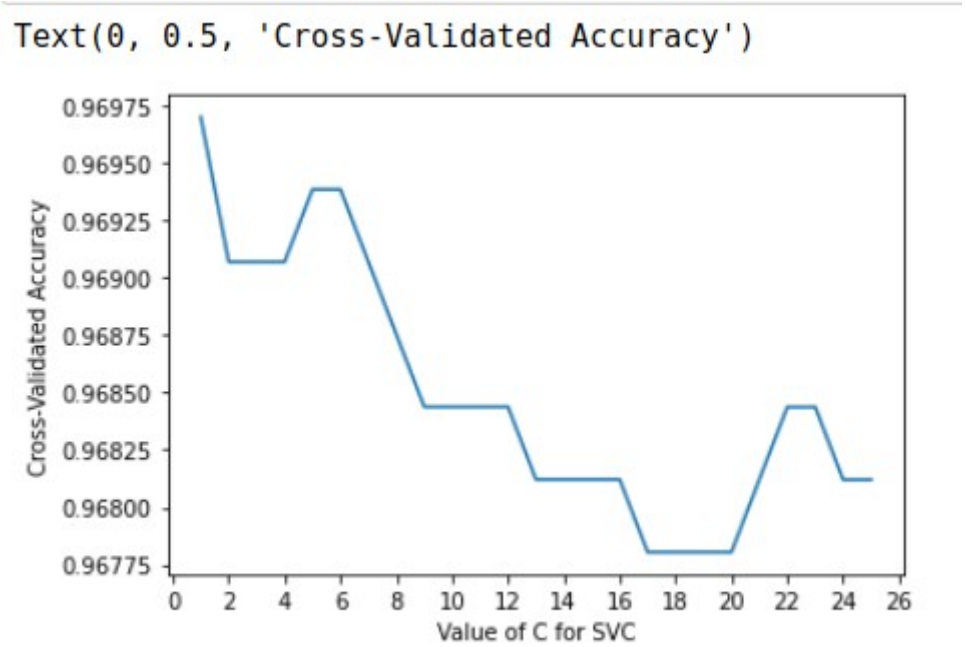
Voice Gender Recognition using Support Vector Machine with different kernals like linear, RBF, poly and hyperparameters C, gamma, degree.

- Input csv file was put into separate “input” folder inside directory and to check if contents were accessible we used **subprocess – check_output** library.
- All required libraries were imported including **pandas, numpy, seaborn, matplotlib, sklearn**, etc
- csv file was read into a **data frame** and correlation was checked
- **features=21** and **instances=3168** were obtained
- **1584 each male and female** labels were identified
- String data was **encoded** into Integer such that **male=1 and female=0**
- dataset was standardized. **Standardization** of datasets is a common requirement for many machine learning estimators implemented in scikit-learn; they might behave badly if the individual features do not more or less look like standard normally distributed data.
- Dataset was split into **test/train**
- SVM model tested with **default hyperparameters** and an accuracy of **0.9763406940063092** was obtained
- later accuracy with **default kernals** was checked :
  linear kernal = 0.9779179810725552
  RBF kernal = 0.9763406940063092
  polynomial kernal = 0.9589905362776026
- since dataset was small **K-fold cross validation** (K-fold cross validation is a procedure used to estimate the skill of the model on new data. Its a resampling procedure used to evaluate machine learning models on a limited data sample) was performed on all kernals with **cv=10** (cross validation):
  linear kernal = 0.9696991175178692

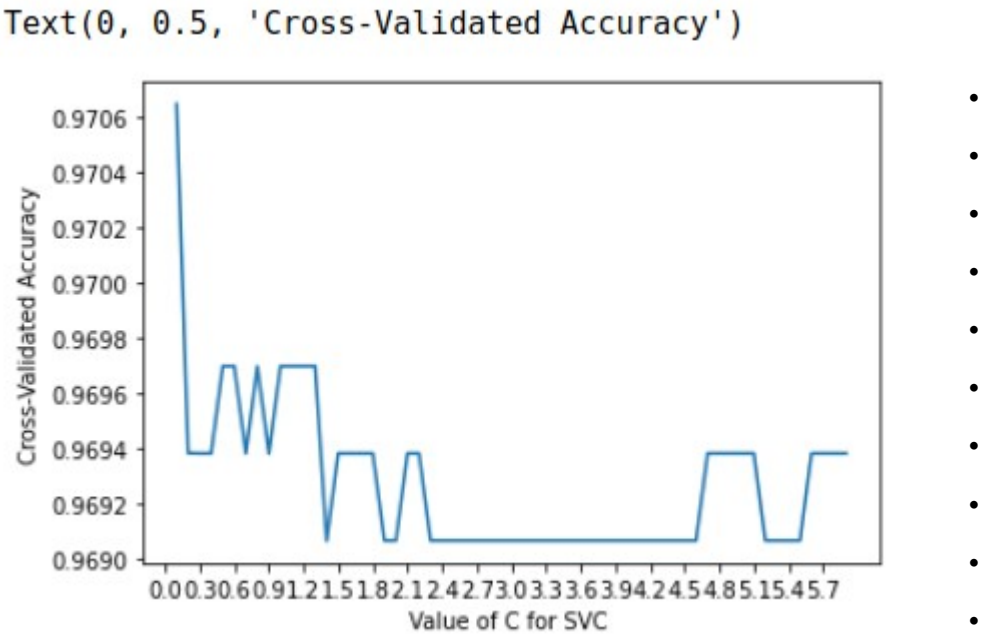
RBF kernal = 0.9665325639899376
polynomial kernal = 0.9450654873617378

- now we started **tuning the hyperparameters**
- first we tested values of **C** (C parameter tells the SVM optimization how much you want to avoid misclassifying each training example) with **linear kernal over a range of 1-26 :**



From the above plot we can see that accuracy has been close to 97% for C=1 and C=6 and then it drops around 96.8% and remains constant.

Hence we teseted again but with **range 0.1-6**



-
- Hence by tuning value of C we can clearly see from the graph above that we get highest accuracy at **C=0.1**
- Simmilarly we tuned hyperparameters gamma for RBF kernal and degree with polynomial kernal getting the conclusions :
  gamma parameter for RBF kernal best value = 0.01
  degree parameter for polynomial kernal best value = 3.0
- thus we finally trained SVM model and performed K-fold cross validation (k=10) with :
  Linear Kernal with C = 0.1
  RBF kernal with gamma = 0.01
  Polynomail kernal with degree = 3
- we finally used the **sklearn.grid_search – GridSearchCV library** to find **best parameter**

```
print(model.best_params_)
```

```
{'C': 0.9, 'degree': 3, 'gamma': 0.05, 'kernel': 'poly'}
```

```
y_pred= model.predict(X_test)
print(metrics.accuracy_score(y_pred, y_test))
```

```
0.9589905362776026
```

which as we can see is :

Ploynomial Kernal with gamma = 3 which gave an accuracy = 0.9589905362776026